

HELLO
MS-DOS 5

EXE

JULY 1991

VOL 6

ISSUE 2

The Software Developers' Magazine



*Artificial Intelligence:
Still not entirely satisfactory...*

*The debate about the limits of AI,
From Alan Turing to Roger Penrose.*

*Constraint programming:
How to make the trains run on time.*

Advanced DIY Modula-2 multi-tasking.

*486 tricks?
New MASM has 'em.*

The Leading Light in Windows Training

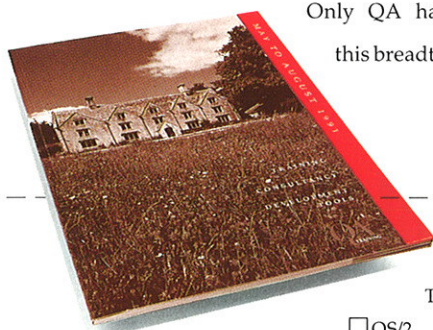


Training has long been recognised as the leader in Windows Training. The world's first company to offer Windows programming courses. The first authorised by Microsoft. And the very first to give technical training in Windows 3.

Our unique range of courses – Windows training for developers, support personnel and users – are widely acknowledged to be, quite simply, the best.

We also have worldwide involvement in consulting on many of the major Windows projects with corporate developers and software suppliers.

Only QA has this breadth



Introduction to Windows

Use Windows mouse and keyboard controls; Customise the Windows desktop; Work with Windows file and print utilities; Use Windows applications productively; Understand the theory of Windows. *2 Days.*

Supporting Windows

Install Windows and explore related issues; Configure Windows and applications; Use network communications and print facilities; Understand Windows internals – memory multi-tasking. *3 Days.*

Microsoft Windows Programming

Use the MS Windows SDK and tools; Write well structured compatible Windows applications; Understand the Windows API and Windows programming principles. *5 Days.*

Advanced Windows Programming

Understand application structure and design techniques in a non-pre-emptive multi-tasking environment. Understand the more complex aspects of MS Windows programming. Write MS Windows using advanced functions and techniques. *5 Days.*

of experience in every aspect of the Windows environment. Experience which is your assurance that a QA Windows course, is certain to maximise

your skills and effectiveness.

For more information, please complete and return the coupon today or call Samantha Trinder on 0285 655888.

Mail to QA Training Ltd, Cecily Hill Castle, Cirencester, Gloucestershire GL7 2EF. UK.

Tick here for full details:

- ☐ OS/2 ☐ Unix ☐ Windows
☐ C ☐ Object-Oriented ☐ Networks
☐ Consultancy ☐ Products

NAME _____ TITLE _____

COMPANY _____

ADDRESS _____

TEL. NO. _____

EXE 7/91



The Leading Light in Windows Training

Editor: Willie Watts
Editorial Assistant: Wendie Leek
Design & Layout: Katerina Adams
 Mark English
Group Advertising Manager: Sandra Inniss-Palmer
Assistant Advertising Manager: Ed Butcher
Advertising Executives: Chris Haill
 Marc Warren
Reader Services & Promotions: Helena Adams
Publisher: Jon Howell
Administration: Rena Gibbs
& Subscriptions: Sandy McDonnell

Cover Photograph Copyright © BBC

Special thanks to Tom Baker.

General

EXE Magazine is independent and not affiliated to any vendor of hardware, software or services. It is published by Process Communications Ltd, 10 Barley Mow Passage, Chiswick, London W4 4PH.

Tel: (Advertising/Editorial/Production) 081 994 6477

(Subscriptions) 044282 4501

Facsimile: 081 994 1533 Telex: 8811418 SPACES G

ISSN: 0268-6872

Subscriptions

EXE Magazine is a monthly journal for software developers. It is available only by subscription, at a cost of £35 per annum (11 issues) in the UK. A subscription implies that this journal will be sent to the subscriber until one of the three expires - AG Macdonell. The magazine is published around the 26th of the month preceding the cover date. There is no January issue. To subscribe or obtain details of overseas/academic rates, please call 044282 4501, or write to The Subscriptions Manager, EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We can invoice your company or take ACCESS and VISA cards. In case of a query about your subscription, please call 044282 4501.

Back issues are available at a cost of £3.50 each - call our Chiswick office (081 994 6477) for a list of issues still in print. We also offer a range of magazine binders, disk holders etc - please call for details.

Editorial

Editorial enquiries should be addressed to The Editor, EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. We welcome letters, opinions, suggestions and articles from our readers. Please write for a copy of our Contributors' Guide. Information contained in EXE is believed to be correct. If errors are found, we will endeavour to publish a clarification in the next available issue.

From time to time, we offer to copy the PC program code described in an article onto diskette for our readers. In this case, please send a blank, formatted disk with a self-addressed, prepaid mailer to the editorial address given above. We can copy both 5.25" and 3.5" disks.

The publishers can accept no liability for any consequences of using software distributed in this way.

Advertising

If you are interested in advertising in this magazine, please write to the Display Advertising Manager/Recruitment Sales Manager (as appropriate), EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH, or call 081 994 6477 for details of our advertising rates.

Pronunciation

The name of EXE Magazine is pronounced to rhyme with 'not sexy magazine'.

Copyright

Material published in EXE Magazine is copyright © Process Communications Ltd. Articles (or parts of articles) may not be copied, distributed or republished without written permission from the publishers.

Issue theme: Artificial Intelligence

ARE THERE LIMITS TO AI?

If we were as smart as Dr Who, could we produce software Cybermonsters? Some AI experts say not, as Tony Dodd explains.

14

ALL CONSTRAINT IS NOT EVIL...

Constraint programming is an AI technique which is descending among mortal programmers. Al Roth and Stuart Watt tell us how it works.

21

MS-DOS V5.0 - A QUANTUM LEAP OR JUST CATCHING UP?

Mark Hamilton blackens his fingers on the interesting techie bits inside Microsoft's new MS-DOS 5.

28

NOW AVAILABLE FOR UNIX

Not content with MS-DOS, OS/2, Windows and the Macintosh, Zortech's C++ has now arrived on John Cant's UNIX machine.

33

MASM 6 - ASSEMBLY LANGUAGE OR NEW 3GL?

MASM 6 is fast as greased lightning; but they've also done some interesting things with the syntax, says Don Binns.

39

3D COMPUTER GRAPHICS - THE FLOATING HORIZON ALGORITHM

You can also use this technique to display contour landscapes, but Graeme Webster continues his 3D series with an elegant Mexican hat.

47

MS-DOS MULTI-TASKING WITH TOPSPEED MODULA-2

Dr Pickard's multi-thread article on Modula-2 and multi-tasking.

56

SOAPBOX

Why Al Roth foams when he hears them say 'AI is dead'.

2

NEWS

The Editor has fallen in love with some sound software, and what Borland, JPI, Microsoft and Zortech got up to at Software Tools.

4

LETTERS

We have some 3D illustrations, plus low-level debugging problems.

10

MAYHEM

Jules May has finished his computer programmer apprenticeship, but what about the young blood who must follow him?

71

THE THIRD SIDE

T_EX is a cheap DTP-beating language from Donald Knuth himself. Peter Flynn guides it through the Triangle test.

72

THE CODE PAGE

Barry Thomas presents a simple PostScript driver.

79

UNIX REGULAR

Seasoned programmer Peter Collinson delves into the UNIX library.

82

BOOKS

In which ex-.EXEite Dan O'Brien toys with two theatrical titles.

88

CROSSWORD

Loyal crosswordist Eric Deeson presents another loyal crossword.

95

STOB

Restored to full health, Verity has a job interview.

96

Sticking up for AI

Artificial Intelligence is hard to define, hard to see and hard to develop. Al Roth argues that these qualities are what make the technology so intriguing.

I wish I had a fiver for every time I ever heard the question 'Is AI dead?' I certainly wouldn't be writing articles like this. I'd be living in an island paradise surrounded by mermaids and consuming my own body-weight in Pina Colada. Still, no smoke without fire... So is AI dead?

Before jumping into this can of worms (*shurely 'opening' - Ed*), it is worth standing back and asking the basic question 'What is AI?'

My favourite answer to this, out of the many that I've heard over the years, claims that AI is 'what we can't do yet.' As soon as we know how to build something, then ceases to be AI. This definition has the advantage of being entirely unassailable. It is also rather useless.

An alternative kind of thinking can be found in the world of chess-playing machines. It was once claimed (I suspect around Turing's time) that if a machine could play a *damn* fine game of chess, then it would surely have oodles of that enigmatic quality - AI. Don't know about you, but the noddier of software can whip me every time and I used to be considered a dab hand at the odd Nimzo-Indian defence - though nowhere near the level of Master, and certainly not on the calibre of a Grand Master, such as the Deep Thought computer trounces every other week. The truth is that modern chess programs do extremely well just using look-ahead search strategies and algorithmic evaluation functions. The AI just isn't there.

So perhaps a chess program that could explain its position in terms that human beings find sensible would have AI. If instead of 'I got the most points' our chess program 'knew' strategic things like 'Centre control is a good idea' as a general declarative statement about what represents a favourable position, we would be more impressed. But even then we would know little more about what 'intelligence' really was. Furthermore, we would probably be falling into the same trap of again moving the goal-posts, believing that AI is what we don't yet know how to do - see definition 1.

So we need to find a workable definition even if only for the time being. The most useful definition I ever heard is that AI is the study of making machines do things that would require intelligence if those things were done by humans. Getting there. The trouble is that it still doesn't really capture the popular position. Most people

read 'Expert Systems' as synonymous with AI. The truth is that expert systems really only represent the first commercially available spin-off for AI technology. What will be the next?

Industry observers have pointed to dwindling attendance at big AI shows. (Especially the ones organised by the American Association for Artificial Intelligence or AAAI). This is offered as clear evidence that AI is going belly-up. At the same time, attendance at

more specialist conferences to do with vision, neural networks and natural language applications (to name but a few) has been steadily increasing. So AI is splintering into more specialist fields. But this is probably to be expected. Just because Science is taught in Schools as Physics, Chemistry, Biology etc does anybody want to argue that Science is Dead?

The point is that AI is now emerging from the lab and into the real world. Languages like LISP and Prolog are trying to shed their AI mantle and demonstrate their use in the real-world as development tools for delivering serious commercial applications. This is the main message of conferences such as Eurol and the planned Practical Applications of Prolog Conference (should appear in the spring of 1992).

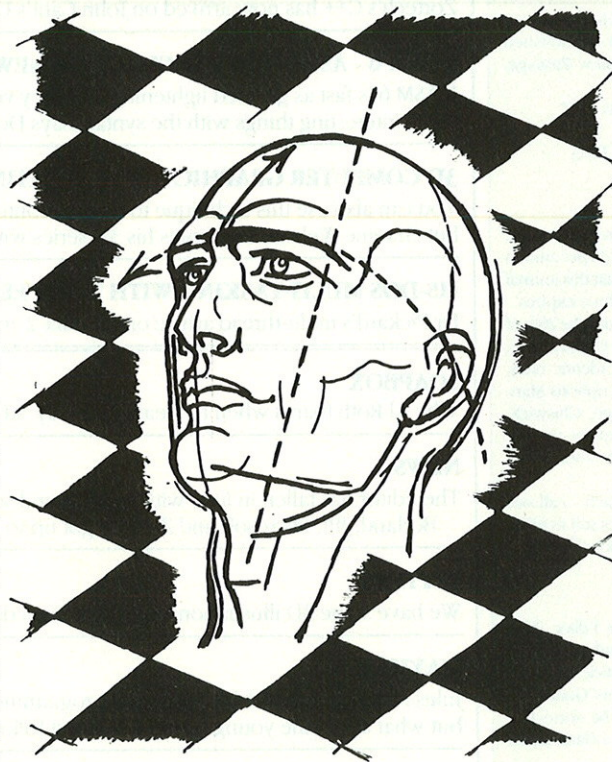
I know of numerous applications that are built using LISP and Prolog that once might have been called

AI, but now can best be viewed as commercial applications/techniques. AI hasn't disappeared, it just allowed some of its techniques to be embodied within classical computer science. I suspect this happens a lot.

For many AI professionals the focus now is on using AI techniques within specific vertical markets - banking, finance, design, manufacturing, environmental monitoring and so on rather than selling AI as the generic solution to all of Mother Earth's problems. For these reasons I would conclude that AI is not dead. Instead it is alive, and well, and living inside new generations of intelligent applications.

EXE

Al Roth is a freelance writer, technology consultant and professional Blackpoolian. He is involved in the organisation of the Eurol and Practical Applications of Prolog conferences.



THE C LANGUAGE

New Timestep C & C++ V3 now available. New Hi-Tech C 386 PDS generates DOS and 386 code, includes own DOS Extender. Refer also to the C++ section.

C COMPILERS

Microsoft C v6	OS/2&PC-DOS	£222
Timestep C for OS/2	OS/2	£335
Watcom C v8 Prof	OS/2&DOS	£310
Avocet AVCOM 86C	PC-DOS	£210
Aztec C86 Professional	MS-DOS	£120
Aztec C86 Developer	MS-DOS	£175
Aztec C86 Commercial	MS-DOS	£250
HIGH C v1.61	MS-DOS	£390
HIGH C 386 Global	PL386&MS-DOS	£650
Hi-Tech C	MS-DOS	£165
Hi-Tech C 386 PDS	PC-DOS	£385
Microsoft QuickC v2.5	PC-DOS	£ 50
M'soft QuickC/QuickAsm	PC-DOS	£ 83
MIX C	MS-DOS	£ 20
MIX Power C & lib s'ce	PC-DOS	£ 30
Timestep C Std	PC-DOS	£110
Timestep C Prof	PC-DOS	£165
Watcom C v8 Std	MS-DOS	£265
Watcom C/386 Std	PL386&MS-DOS	£585
Watcom C/386 Prof	PL386&MS-DOS	£780
Avocet C Avcom Z80	CP/M-80	£210
Aztec C Personal 1.06D	CP/M-80	£120
Aztec Commercial 1.06D	CP/M-80	£200
Hi-Tech C	Z80 + CP/M-80	£105
Mix C	Z80 + CP/M-80	£ 35
Aztec C68K MPW C	MACINTOSH	£110
Aztec C68K C	MACINTOSH	£ 75
Aztec C68K/AM Developer	AMIGA	£175
Aztec C68K/AM Prof'snl	AMIGA	£120
Aztec C68K/ST Developer	ATARI	£175
Aztec C68K/ST Prof'snl	ATARI	£120
Hi-Tech C 68000	ATARI	£140
Mark Williams C v2	ATARI	£110
Laser C	ATARI	£135
Prospero C	ATARI	£ 80

DISK COPYING SERVICE

We can copy files to and from 600 disk formats including CP/M, CP/M-86, MS-DOS, PC-DOS, ISIS, APPLE, SIRIUS, BBC, TORCH, APRICOT, HP-150, TRSDOS, DEC RT-11, IBM BEF, ATARI ST, AMSTRAD, MACINTOSH.

Our charge is £10.00 + disk + VAT with discounts on small quantities and disks are normally despatched within 24hrs of receipt.

For more information call us.

PROGRAM EDITORS

BRIEF for OS/2 & DOS	OS/2&DOS	£199
Kedit v4 for OS/2	OS/2&PC-DOS	£105
Epsilon v5 for OS/2	OS/2	£135
QEdit Advanced for OS/2	OS/2	£ 49
MKS Vi for OS/2	OS/2	£130
SPE for OS/2&DOS	OS/2&PC-DOS	£255
EC Editor v2.1	PC-DOS	£ 40
Epsilon v5	PC-DOS	£135
Kedit v4	PC-DOS	£ 95
ME v2.01 (w.C source)	PC-DOS	£155
MIX Editor	MS-DOS	£ 20
MKS VI	PC-DOS	£ 99
Multi-Edit v5	PC-DOS	£ 70
Multi-Edit Professional	PC-DOS	£115
Norton Editor v2	PC-DOS	£ 45
QEdit Advanced	PC-DOS	£ 45
RED v6.6 (with C S'ce)	PC-DOS	£ 65
SPE for DOS	PC-DOS	£175
Syndie	MS-DOS	£445
XTC (with Pascal source)	PC-DOS	£ 50
Vedit-Plus v3	PC-DOS	£125
Vedit-Plus (CRT)	MS-DOS	£120
Vedit-Plus	CP/M-86	£120
MIX Editor	Z80 + CP/M-80	£ 25
Vedit-Plus v2.33	CP/M-80	£110

For more information please call us.

PRICES & DELIVERY

Prices do not include VAT or other local taxes but do include delivery in UK and Europe.

Please check prices at time of order, ads are prepared some weeks before publication.

This page lists some of our products. Call us for a complete pricelist.

Order by phone with your credit card.

C CROSS COMPILERS

We supply 2500AD, Avocet, Aztec, Lattice, IAR and Hi-Tech Cross Compilers hosted on MS-DOS and targeted on Z80, 6502, 6801, 68HC11, 6301, 6809, 7811, 8051, 8096, 68000 & 68020. Please call for information or advice.

C INTERPRETERS

C-terp V3.0X	PC-DOS	£165
Instant C/16M	PC-DOS	£700
Interactive C	PC-DOS	£195
Introducing C	PC-DOS	£ 85
RUN/C Professional	MS-DOS	£110

C LIBRARIES & TOOLS

COMMS LIBRARIES

C Asynch Manager V3's'ce	PC-DOS	£135
Essential Comms(MS&TC)	PC-DOS	£180
Greenleaf Comms(source)	PC-DOS	£195
Multi-Comm (MS)	PC-DOS	£190
Silvercomm C Asynch	PC-DOS	£140

DATABASE/FILEHANDLING

Btrieve for OS/2	OS/2	£360
Btrieve for Windows	MS-DOS	£360
Btrieve v5.1	MS-DOS	£360
XQL	OS/2,PC-DOS	£440
Xtrieve Plus	OS/2,PC-DOS	£360
C/Database T'chest s'ce	PC-DOS	£ 39
CBTREE	(source any C)	£135
C-Index/Plus	(source any C)	£175
C-ISAM (L,MS)	MS-DOS	£170
C-tree	(source any C)	£300
C-tree Plus	PC-DOS	£360
Essential Btree (s'ce)	PC-DOS	£110
CQL s'ce	PC-DOS	£270
Faircom Toolbox Prof v2	PC-DOS	£680
Pro-C & Workbench v2	PC-DOS	£585
R-tree	MS-DOS	£140
db-VISTA DBMS Object	PC-DOS	£450
Lattice dBCIII	L,MS) MS-DOS	£185
Lattice dBCIII+	OS/2&MS-DOS	£365
StfCus Btree&Isam	(see any C)	£ 80

GRAPHICS

Essential Graphics v3	PC-DOS	£215
GFX Graphics	PC-DOS	£ 80
GraphiC v5 (MS)	PC-DOS	£245
graphics MENU/META v3	PC-DOS	£125
GSS Kernel system	OS2,PC-DOS	£425
GSS Graph Dev T'kit	OS2,PC-DOS	£425
GX Graphics	PC-DOS	£ 95
HALO v3 (MSC5,TC)	PC-DOS	£215
HALO Professional (C)	PC-DOS	£335
HGraph (MSC)	PC-DOS	£ 80
Ingraf (MSC)	PC-DOS	£185
MetaWINDOW v3.7b	PC-DOS	£145
MetaWINDOW/Plus	PC-DOS	£195
TurboWINDOW/C (Turbo)	PC-DOS	£ 85
QuickWINDOW/C (QC)	PC-DOS	£ 85

LOW PRICES

FOR MICROSOFT & BORLAND PRODUCTS

Microsoft Basic PDS 7.1	222.00
Microsoft C PDS 6.0	222.00
Microsoft C/Windows SDK	364.00
Microsoft Cobol 4.0	365.00
Microsoft Fortran 5.0	205.00
Microsoft Macro Assembler 6.0	63.00
Microsoft Pascal 4.0	139.00
Microsoft QuickBASIC 4.5	50.00
Microsoft QuickC 2.51	50.00
Microsoft QuickC & Assembler	83.00
Microsoft QuickPASCAL 1.0	50.00
Microsoft Windows 3.0	70.00
Microsoft Windows SDK	265.00
Borland C++ 2.0	206.00
Turbo C++	48.00
Turbo Debugger & Tools 2.0	89.00
Turbo Pascal 6.0	69.00
Turbo Pascal Professional 6.0	137.00
Turbo Pascal for Windows	103.00
Object Vision 1.0	69.00
Paradox 3.5	409.00
Paradox Engine	223.00
Paradox MultiPack	615.00

WITH FULL TECHNICAL SUPPORT

PROLOG LANGUAGE

Arity/Prolog Comp	OS/2	£745
ADA Educ.Prolog	MS-DOS	£ 26
ADA FS Prolog	MS-DOS	£ 45
ADA VMI Prolog	MS-DOS	£ 70
ADA VML Prolog	MS-DOS	£125
Arity/Prolog Comp. v6	PC-DOS	£520
Arity/Prolog Int. v6	PC-DOS	£220
Arity Standard	PC-DOS	£ 75
Flex Expert Dev Sys	PC-DOS	£1210
LPA PROLOG Prof v3	PC-DOS	£755
Micro-PROLOG v3.1	MS-DOS	£ 70
PDC Prolog	PC-DOS	£240
Micro-Prolog v3.1	CP/M-86	£ 70
Micro-Prolog v3.1	CP/M-80	£ 70
Prolog-1 v2.2	CP/M-80	£225
LPA Mac Prolog v3 Apple Mac Pls		£450

SCREEN & WINDOWS

C-Scape +Look&Feel	OS/2&PC-DOS	£390
Vitamin C for OS/2	OS/2&DOS	£240
C View Manager (s'ce)	PC-DOS	£245
Curses/PC w. s'ce	PC-DOS	£235
Entelekon Windows(s'ce)	PC-DOS	£105
Greenleaf Datawindow	PC-DOS	£215
Greenleaf Makeform	PC-DOS	£ 75
Lattice Curses V Lib	PC-DOS	£105
Multi-windows (MS,L)	PC-DOS	£190
Panel Plus II (source)	PC-DOS	£260
Power Screen (MS&TC)	PC-DOS	£ 99
Vermont Views (MSC,TC)	PC-DOS	£345
Vitamin C	(source) PC-DOS	£145

GENERAL FUNCTIONS

C/Utilities T'chest s'ce	PC-DOS	£ 39
C-Worthy w CW Architect	MS-DOS	£410
C Tools Plus/6.0 (MS)	PC-DOS	£105
CodeRunner	PC-DOS	£ 95
Entelekon Funct(source)	PC-DOS	£105
Essential C Utilis v5	PC-DOS	£140
G'leaf Fin'cial Mathlib	PC-DOS	£215
G'leaf Super Functions	PC-DOS	£160
Greenleaf Functions s'ce	PC-DOS	£125
Lattice C Comp Kit+s'ce	PC-DOS	£160
PDK£1	PC-DOS	£ 70
Power Search	PC-DOS	£ 99
Resident-C (MS,L)(s'ce)	PC-DOS	£135
Silverware C EMM Lib	PC-DOS	£105
The Heap Expander(MS&TC)	PC-DOS	£ 65
Turbo C Tools v2	PC-DOS	£ 95
WKS Library	PC-DOS	£160

MATHEMATICAL LIBRARIES

C Lang Sci Lib v2	ANY C	£160
Mathpak 87 (L,MS)	MS-DOS	£ 80
Science & Eng Tools	MS-DOS	£ 59

TOOLS & SUNDRIES

PC-Lint	OS/2,MS-DOS	£ 80
Clean+ for C	PC-DOS	£140
Databoss (C)	PC-DOS	£320
Lattice Comp Companion	MS-DOS	£ 70

REAL TIME & MULTI-TASKING TOOLS

Concurrent C (PC/MPX)sce	MS-DOS	£ 60
Multi-C (MSC,TC)	PC-DOS	£190
Op.Sys.T'box (MS) s'ce	PC-DOS	£ 69
Timeslicer v6 (MS6)	PC-DOS	£200
Over-C (MS)	PC-DOS	£225

PROGRAMMING TOOLS

Ada Compilers	Algol Compilers
Assemblers & Libs	AWK
Basic Compilers	Basic Interpreters
Basic Utilities	Basic Libraries
BCPL Compilers	C Compilers
C Interpreters	C Libraries
C Utilities	Cobol Compilers
Comms.Libraries	Cross Assemblers
Database Libs.	Debuggers
Dis-assemblers	Editors
Engineers Libs.	Expert Systems
Forth	Fortran Compilers
Fortran Libraries	Graphics Libraries
Icon	Linkers/Locaters
Lisp	Modula-2
Nial Interpreters	OPS 5
Pascal Compilers	Pascal Libraries
Prolog	Rexx
Risc	Screen Libraries
Smalltalk	Snobol

We stock many items for which there is no space in these advertisements.

GREY MATTER
Prigg Meadow, Ashburton, Devon TQ13 7DF. Fax: (0364) 53071
TEL: (0364) 53499

GREY MATTER
Prigg Meadow, Ashburton, Devon TQ13 7DF. Fax: (0364) 53071
TEL: (0364) 53499

GREY MATTER
Prigg Meadow, Ashburton, Devon TQ13 7DF. Fax: (0364) 53071
TEL: (0364) 53499

Chips are down

Intel has announced reductions in the prices of all its maths coprocessors. A top-spec 33 MHz 387 DX has been reduced from £379 to £176.60, a 16 MHz 387 SX from £199 to £88.10. I obtained the old prices - which were omitted from Intel's press release - from a Powermark Ltd catalogue (081 951 3355), which also lists the prices for extra-featured IIT/CYRIX/ULSI clones. These are £299 and £155 respectively. Could there be a connection here?

OS/2 Idle

The OS/2 User Group has produced Triplecheck, a set of three utilities for OS/2 developers. MON is an idle time monitor, enabling the programmer to tune his use of OS/2's multi-tasking. SWAPMON monitors the swap file size and can also be used to spot memory leakages. CLOCK displays the current time. Triplecheck costs £50 to non-members; details on 0285 655888.

Virus Bergerac

This office's favourite source of considered Virus information, the Virus Bulletin, is holding a conference on the island of Jersey from 12th to 13th of September. VB has attracted a strong collection of experts who will lecture on such diverse subjects as 'The Bulgarian and Soviet Virus Factories' and 'Disassembly, Forensics and Recovery'. The registration fee is £595, phone 0235 531889 for the literature and booking form.

286 PS/2 upgrade

SX/Now! from US-based Kingston Technology is a 386SX upgrade board for IBM PS/2 Models 50, 50Z and 60. It plugs directly into the 286 socket, providing a 16 KB memory cache, an on-board clock and a choice of 16 or 20 MHz 386SX processors. The UK distributor Datrontech (0252 313155) will sell you these boards for £475 and £495 respectively. Versions for other 286 machines are planned.

Dan's parting shot

A particularly good crop of wrong/missing telephone numbers last issue. Here's a list of corrections and scapegoats: QBS (Dozy News Editor gave a Fax number) 081 994 4842, UKIC (Dozy News Editor transposed a digit) 071 269 3159, EQ Consultants (Half-wit Editor failed to read note from Dozy News Editor) 0334 84248 and IntelliCorp (blaming this on BT) 0962 735348. Mr Dan O'Brien has now departed .EXE for the bright lights of comedy; his fans may keep in touch by listening to BBC Radio 4's Week Ending program. We wish the rat the best of luck.

How did the show go?

Thanks for asking. Software Tools '91 (as sponsored by .EXE Magazine, actually) went very well. Microsoft - our co-sponsor of the PC development area, bless them! - was there in force, showing off Visual Basic (see last month) and MS-DOS 5 (see Mark Hamilton's article this month for some techie details). Lucky punters, who smiled nicely at Microsoft UK Languages Supremo Andrew King, may have glimpsed a very slick-looking Windows-based QuickC (my name for it, product details and release date not yet available), incorporating a parsing editor - as last seen in the otherwise limp QuickPascal - and a code generator for producing the framework of Windows programs. No sight nor smell of C++ yet, though.

On the Borland stand they were, for once, a little bit on the defensive, thanks to the acclaim that Visual Basic was attracting. Borland US's Gene Wang assured me that the next release of Borland C++ would knock anything that Microsoft could do into a cocked hat. As well as incorporating the ObjectWindows class library, he explained, eyes glinting, there would also be a new GUI drawing tool which would, among other things, enable the punter to produce applications with the smart 3D look which characterises Turbo Pascal for Windows. More details of this next month, I hope.

If the big compiler companies had one eye fixed on the future, the slightly smaller ones had their hands full in the present. Friends of JPI were relieved to see that the company's long-trailed V3.0 compilers, now including C++ and OOP Pascal, finally put in a physical appearance. JPI's Top-Speed compilers are based on a modular system; you buy the IDE separately, then bolt in whichever language engines you require. JPI is pushing the multilingual aspect of the system hard - it hopes to attract, for example, Pascal programmers who wish to access the wealth of third party libraries written for C. All modules cost £59; MS-DOS and OS/2 is supported, plus Windows with an SDK.

Zortech, the C++ specialist, was also launching V3.0 packages, although it was a frustrating 'one manual short of the set' away from being able to sell the software

on its stand. The stuff should be available by the time you read this. Zortech C++ V3.0 is a very aggressively priced and feature-packed range of software. The entry-level £249.95 box, C++ for Windows and DOS, contains all you need to write Windows programs (ie no SDK). A new WINC library allows DOS command line programs to be quickly converted to Windows; Zortech says it has tidied up all the little bugs which dogged its V2.1 compiler under Windows. There is also a bundled royalty-free 16-bit DOS extender, plus debugging support for the lot. The Developers Edition (£399.95) adds a royalty-free 32-bit DOS extender, a 32-bit version of the compiler, OS/2 support, debugging support for the extra environments, a C++ Tools Library and source for the standard libraries. The Science and Engineering Edition (£599.95) adds Weitech coprocessor support, the M++ array library and IEEE 748 standard floating point. Zortech hopes to attract FORTRAN programmers into the world of C++ with this package. Incidentally, if you are looking for a bargain, you could do worse than contact System Science, which will throw in the Zortech C++ Video Course - normally priced around £300 - with any purchase of Zortech V3.0 before the end of August.

While we are on FORTRAN (contrived link) NAG launched what it claims is the world's first FORTRAN 90 compiler. This is both written in C and generates that language as its output, for maximum portability. The compiler is being offered on a range of workstations - contact NAG for details.

There were lots of other interesting and good things going on at the show, so my apologies to everybody I failed to meet (especially Ms Fau, representing Programming Research, who apparently went bananas trying to track me down) or I have omitted from this summary. As to all those who enquired warmly after Ms Stob at the .EXE stand: Verity says she is sorry she couldn't make it, and she sends her love and kisses. And no, Mr S of Basingstoke, she regrets she is already spoken for.

Telephone numbers: Microsoft 0734 500741, Borland 0734 320022, JPI 0234 267500, Zortech 081 316 7777, System Science 071 833 1022 and NAG 0865 511245.

Four Monitor Cebra

Bournemouth-based Cebra Communications has released a new version of its Windows 3 driver. With two of Cebra's Dual VGA boards in your machine, you can now get Windows to run on four monitors simultaneously. Cebra goes on about its uses in Industrial Control and Dealing Room systems, but the application that strikes me is the possibility of creating a really dramatic display on your stand at trade shows. Cebra's boards are available for both AT-bus and MCA - call 0202 299048 for pricing info.

WELL? IT DOES EVERYTHING ELSE

Clipper is the standard for companies all over the world who are serious about selecting the correct software to develop their applications. From accounting to zoology, from finding oil to finding your way home, Clipper can help any company to do what it needs to do quicker and more efficiently. Clipper combines the power of a programming language with the convenience of a data base to offer large companies a means of saving money. And offer small ones a means of making it. However, there are some things that Clipper can't do. We'd be straining credibility to suggest it would make the tea for you. Although out of 300,000 programmers using Clipper, there just might be someone out there . . .

 **Nantucket**®

Clipper 5.0

The Application Development Standard

NOW SHIPPING 5.01

Nantucket UK Limited

8 Bridgegate Centre, Martinfield,
Welwyn Garden City, Herts. AL7 1JG
Telephone: (0707) 373600
Fax: (0707) 373350

I'd like a flavour of Clipper, please send me details.

Please tell me about the uses Clipper is being put to

Name _____ .EXE 7/91

Job _____

Company _____

Address _____

Phone _____

CIRCLE NO. 962

Pascal Language Forum

I don't remember hearing of the non-profit Pascal Language Forum, which has sent me details of a Conference it has just held. Rather than titillate you with what you missed, here is the contact information for membership, in case it decides to hold another. Write to the Administrator, Pascal Language Forum, PO Box 30, Fareham, HANTS PO16 8LZ or call 0329 233038.

Sourcerer's Apprentice

Sourcerer's Apprentice is a new version control system which majors on close links with the BRIEF Programmer's Editor. It costs £175 for a personal edition, £429 for a professional version, from Solution Systems (0763 244141).

XENIX Update

SCO has shipped XENIX 386 Release 2.3.4, 'underscoring its long-range commitment to users of its SCO XENIX family of operating systems', as the company puts it. New features include on-line documentation, the Korn Shell, improved SCSI drive performance and faster installation. The software costs £450 for a two-user licence and £675 for an unlimited-user licence.

MAGIC Port

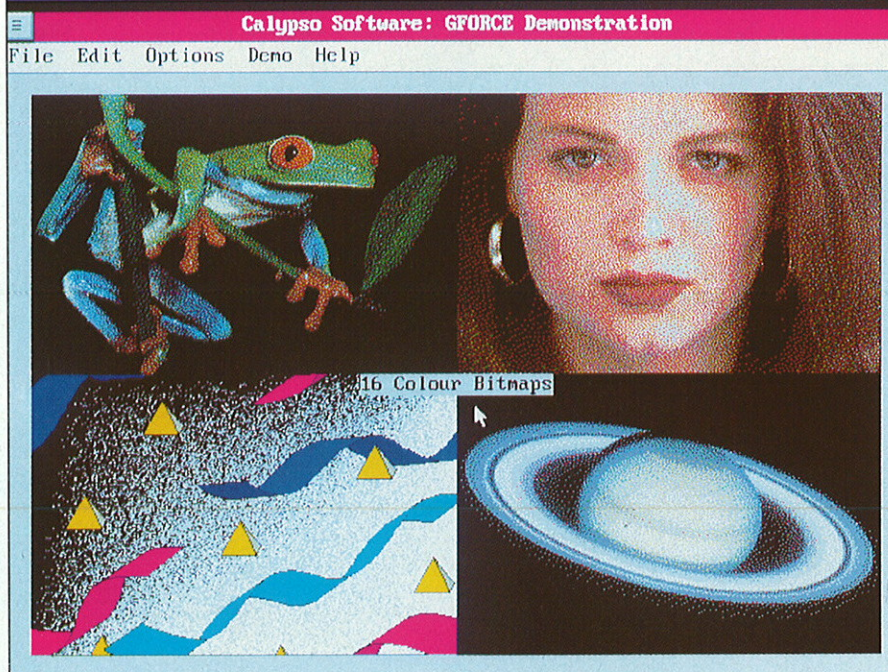
MSE has ported its MAGIC application generator to VAX/VMS and SCO UNIX V platforms. MAGIC is based on tables - there is no conventional procedural programming language. The manufacturer makes the usual claims of enhanced productivity, ease of maintenance etc that are always made for application generators and 4GLs. However, one can see that the ability to port an application between MS-DOS and the larger-machine platforms might be attractive. Prices range from £695 for low-end MS-DOS versions up to £3300 for UNIX and VAX. MSE is on 081 902 8998.

Oh no!

'We are releasing yet another Dongle', writes Glyn Williams of User Friendly Microsystems (0527 510 105), correctly anticipating your correspondent's initial reaction. He is stressing Everkey II's 'Single Wire Zero Load', which should fix any laptop/notebook PC compatibility problems that dog its brother dongles. There are other goodies too: Execution-count and Expiration Date Limits for demos, reprogrammable locks, protection of programs without access to source and 16 bytes programmable RAM.

GFORCE GUI

My illustration shows a screen produced by a new Clipper library from Calypso Software called GFORCE. Written in assembler and C, GFORCE provides the Clipper programmer with a Windows-like GUI. There are 3D buttons, pull-down menus, dialogue boxes, icons, fonts and full mouse control. The package comes with Clipper source for a font editor and an icon editor. Bitmaps are handled as Clipper character strings, so can be manipulated with functions such as STRTRAN() and SUBSTR(), and PCX files may be imported via a supplied utility. The software is compatible with both Summer '87 and 5.0 releases of Clipper and costs £185. The distributor is QBS Software, contact 081 994 4842.



Brill! Mega! Fab!

What is this? In the .EXE office, a brand new Virtual Reality machine lies ignored, still clad in its polystyrene packaging. In a corner, a Cray gathers dust. Everybody is crowded around the Editor's PC, pushing and jostling to have a go...

To begin at the beginning. A year or two ago, a couple of engineers, working in the capital of Northern England, designed a small embedded system incorporating a digital sound reproduction facility. In the course of building it, these Geordies completely sussed the programming of the chips. Then they noticed that their hardware was the same as incorporated into the standard PC-compatible, and realised that they had an opportunity to drag the PC's crummy sound system screaming (*sic*) into the 1990s.

Quantech's SoftSpeak II+ package comprises a microphone, a battery-operated A-to-D unit which plugs into the printer port, a magnificent graphics-based sound editor, a library of sound samples and a rich assortment of support software. SoftSpeak lets you play recorded sound through your PC's speaker. The only limitation is that your PC must clock at 8 MHz or greater, although if the machine is equipped with a particularly poxy speaker, Quantech recommends substituting the car radio one that you ripped out of the

Morris Minor after Geoff wrote it off. The input unit lets you record live sound, and has a second jack input for connection to a gramophone or similar apparatus. Once captured, the editor lets you cut and paste the sample (for M-M-Max Headroom effects), increase the volume, add reverb and echo, change the pitch and generally horse around. The edited sample is then saved to disk, where it may be played back using one of the utilities, or 'compiled' to a stand-alone, self-playing .EXE file. The sound quality is astonishing.

All tremendous fun, but the system does have serious applications. Quantech supplies device driver software, so it is possible to make your own programs speak (Quantech charges a licence fee of a fiver a shot for commercial use of the driver). There is a file indexing system which enables you to pluck out individual words - Quantech demonstrates this with a Speaking Clock program. There's loads more bits and pieces, which really demand a full review, so - faked sigh - I'll just have to keep the demo system cluttering up my desk a little longer.

Quantech may be reached on 091 2280513. SoftSpeak II+ costs £139.90 + VAT (or £59.95 without the recording hardware), which makes it, in the opinion of the writer, the bargain of the year.

If you have to get more out of your next C++ Compiler make sure there's a lot more in it

Environment friendly
TopSpeed's 10 Windowed, fully
configurable environment makes
for light work and light-speed

Make your code mean and lean
TopSpeed Smart-Linking
eliminates unused code for
faster tighter programming

Move into Windows
Support for MS Windows 3.0
and OS/2 Presentation Manager

TopSpeed - to the power of 4
C++, C, Pascal or Modula-2 can be
compiled together using TopSpeed
- the Multi-Language Programming
Environment

Smash the DOS 640k barrier
Overlay Code & Data with TopSpeed

Rogue Wave (tm) Class Library
In a class of its own, with over
50 C++ Math & Tools routines

Putting more into standards
100% AT&T 2.1
implementation of C++

Search & destroy that Bug
Debug fast with TopSpeed's VID

Optimise your code, go TopSpeed!
Fast & compact code every time

Hypertext Help
Context sensitive help system
that you can tailor to meet
the needs of a professional
development team or a student
programmer

Mouse support or Macro-keys
TopSpeed gives you more choice

Project System

Choosing your next compiler is no mean feat. Whether you're just putting a toe in the water or about to embark on a major development project with C++, you'll need to know that the compiler you choose gives you the productive edge.

TopSpeed C++ packs more in

TopSpeed C++ packs more in, whether you are developing for DOS, Windows 3 or OS/2. Develop mean-and-lean code with TopSpeed's super-optimizing code generator, Smart-Linking and Project system. Then, add the most extensive C++

class library available and you too can now harness the real power of C++.

TopSpeed Putting more into standards

TopSpeed C++ not only complies with the industry standards offering 100% conformance to AT&T V2.1, but also packs in a range of extensions to take your programming into hyperspace!

TopSpeed just the environment for C++

To make the most out of the C++ and the power of OOP you'll need the development environment that packs the most in.

TopSpeed's unique Project system frees you to be creative whilst it manages file dependencies, compilation, assembly, linking and much more.

TopSpeed C++ in a family of thoroughbreds

C++ is just one of a family of four compilers that integrate seamlessly into the TopSpeed Multi-language Development Environment. From \$99 per module you can custom-build a development system that takes your programming TopSpeed!

For more information call JPI on the number below.

EUROPE

+44 (0) 234 267500

CIRCLE NO. 963

JPI

Jensen & Partners International

USA

(415) 967 3200

I THOUGHT THIS WAS
QUALITY SOFTWARE-NOW MY SYSTEM'S
JUST
HANGING!



**Don't leave
your customers
HANGING...**



BEFORE YOU SHIP ANY PROGRAM, CHECK IT WITH



BOUNDS-CHECKER™

WINNER
PC MAGAZINE
BEST OF 1990

JANUARY 15, 1991

PAGE 168

**FINDS OUT-OF-BOUNDS MEMORY ACCESSES
AUTOMATICALLY**

Your program may have 10,000 to a million lines of code. It may occasionally hang mysteriously or it may appear to run flawlessly every time. But under DOS, how can you ever be sure that your program is not corrupting memory it does not own? The only way to be 100% sure is to BOUNDS-CHECK before you ship.

To use BOUNDS-CHECKER you build your program with debugging information (we support most compilers including Microsoft, Borland & JPI). Then you just type <BC file-name>. BOUNDS-CHECKER sets up the 386™/486 for protection and lets your program fly. If your program accesses memory it does not own or overwrites its own code, BOUNDS-CHECKER pops up displaying the offending SOURCE-LINE or instruction.

Programming under DOS is a gamble, so why not stack the odds in your favor--CALL TODAY.

(603) 888-2386

Call by 4:00 PM EST TODAY and ask us to EXPRESS you an info packet. In most cases you will receive it by 10:30 AM tomorrow. (USA only)

All Nu-Mega products require a 386, 386SX or 486. MS-DOS and Codeview are trademarks of Microsoft Corp. 386 is a registered trademark of Intel Corp. Nu-Mega, BOUNDS-CHECKER, Soft-ICE and CV/1 are trademarks of Nu-Mega Technologies, Inc.

**Nu-Mega
TECHNOLOGIES INC**

P.O. Box 7780 • Nashua, NH • 03060-7780 U.S.A.
(603) 888-2386 • Fax (603) 888-2465

CIRCLE NO. 964

OTHER FINE NU-MEGA PRODUCTS...

The ultimate systems debugger.

Debug:

- Interrupt routines
- Device drivers
- T&SRs
- ROMS
- Applications
- Overlays

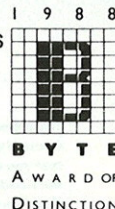
Soft-



Features:

- Break out of a hung program
- Real time Break-Points
- Back-Trace history
- Works with other debuggers

If you are debugging an application, Soft-ICE is seamlessly integrated with BOUNDS-CHECKER so you can easily go back and forth between BOUNDS-CHECKING and debugging: a combination many programming professionals can't live without.



**Run CODEVIEW for Windows on a single monitor
CV/1.....\$129**

- Runs in a window • No annoying flash •
- Runs on any display that supports windows •

BOUNDS-CHECKER\$249
Soft-ICE\$386

Special Offer
Buy BC & S-ICESave \$100

**30 Day
Money Back Guarantee**

Codehigh Amnesty

Codehigh Ltd publishes a floppy-disk database called Inside Information containing an index to product references across 11 computer magazines, including this one. It is a good package, ideal for answering those 'Do you know if you, or anybody else, has reviewed the so-and-so product' questions which we get here about 20 times a day (aha! you've detected a selfish motivation in this story). The database is updated monthly, and is available on a subscription basis.

Codehigh is running a promotion specifically for .EXE readers. The scheme goes like this: you send the company a cheque for £5 made payable to *Amnesty International*, with a note specifying the disk size that you require and your address; it will send you a copy of the search software with a database containing all the product references from the last two years of .EXE Magazine. The database is indexed by product name, supplier and product category.

Write to Codehigh at Harrow Way, Whitchurch, HANTS RG28 7QT, marking the envelope 'Amnesty/.EXE offer'.

Milspec goes Turbo

Coventry-based Milspec Systems, a software house which previously specialised in cheap Ada products, has diversified. Turbo/SDK is an intriguing sounding package: a Turbo Pascal (V5.0 or better) add-on which lets you write Windows 3 programs. You don't need Turbo Pascal for Windows, nor the Microsoft Windows SDK. There is a set of libraries, an icon editor and overlay support.

Standard mode Windows is not supported, but if you are running in enhanced 386 mode, you can run the Turbo Pascal IDE in one window (complete with watches and other debugging aids) while the program runs in another window - no need to obtain a second monitor, or to have to flip to text mode all the time. Surprisingly, for this type of software, it does not use OOP.

Milspec admits that Turbo/SDK is not as sophisticated as Turbo Pascal for Windows,

but given the price-tag of £59, this might not matter. The product is due for release at the end of June. Milspec's phone number is 0203 670770.

Summer Catalogues

The software distributors have all brought out their Summer Catalogues, which can make entertaining free reading. I'm afraid it's too late to join in the prize crossword competition in the back of Real Time Associates' newsletter (first prize: a magnum of champagne, second prize: a (sic) 'Modula-2 is Wirth IT' T-shirt), but if you are loopy about Modula-2, then this is one you should get (RTA is on 081 656 7333).

If it's T-shirts you want, however, call the Software Construction Company (0763 244114) before July 31, and ask the man to send you the gen on Borland's Windows products. He will bung in a handsome T-shirt based on the Borland 'C++ acid trip' motif (as featured on the Turbo C++ box) and a copy of SCC's Toolbox catalogue.

The prize for the flashiest catalogue in my news-box, however, goes hands-down to Software Paradise (0222 887521). The cover looks more like a travel agent's give-away than a software dealer's price list, but the 200 pages are chock-a-block with interesting utilities, specialist libraries, wacky compilers and obscure operating systems.

British Design Awards

You will have to get a shift on, but there is still time to enter for the British Design Awards 1992. The Awards are given by the Design Council, and are split into various categories, including Computer Software. Previous winners include JSB Computer Systems with 'Multiview' and NextBase's famous 'Autoroute' mapping package. There is no charge for entering and the forms are available from the Design Council (071 839 8000). Closing date for entries is 19th July 1991. I reckon Quantech should have a crack at it (see Brill! Mega! Fab! story).

PDOS 4.1

Eyrisoft has announced V4.1 of the PDOS real time operating system. New features include 68040 support (specifically: changes to exception processing and cache handling) and better task scheduling, providing improved throughput and response time. There are extra device drivers for optical disks and other large media, plus the basis for access to all the MS-DOS floppy disk formats - a full MS-DOS file manager is promised 'soon'. The manufacturer says that PDOS is the first real time operating system to run on Force GmbH CPU 40 and Motorola MVME 165 boards. Prices for PDOS range from £400 for a ROMmed version up to around £6000 for a Sun 3 Developer's kit. Eyrisoft may be contacted on 0332 384978.

WNDX

Another portable GUI library for you. Canadian company WNDX's Professional GUI Development System is a C-based event-driven library which provides single source code support across an impressively wide set of platforms: UNIX X, MS-DOS, Windows 3 and (here's the unusual one) Macintosh. Call 0101 403 244 0995 for details.

polyFORTH/32

Comsol, the FORTH company, has introduced polyFORTH/32, an MS-DOS compatible 32-bit 80386 protected-mode implementation of that language. Complete with the £875 package is a multi-user real-time executive which can cohabit with MS-DOS, an integer and fixed point maths package, editing and debugging utilities and complete source code for the lot. If your Reverse Polish Notation is up to it then you can find Comsol on 0932 352744.

PL/M escape

If you have lots of code written in PL/M, and you would rather that it were C, then maybe you should contact Ashling Microsystems in Maidenhead (0628 773070). Ashling's MPS-PLC PL/M to C converters accept PL/M 80, PL/M 86, PL/M51, PM/M96, PL/M286 and PL/M386 (depending on which version you buy) and convert it into K&R C. The programs perform a full syntax analysis and, according to Ashling, require minimal tweaking of source and output. The software runs on PC-compatibles, and costs £455 per unit.

Nu-Mega and CodeView

Nu-Mega has announced V2.0 of its 'give CodeView a bit of help' utility CV/1. The new release lets you run Microsoft's CodeView in 'Small Window Mode', leaving most of the screen free for the program that you are debugging. Price is \$129 plus \$47 shipping. Phone Nu-Mega on 0101 603 888 2386.

Auto-Tutorial

Electrovision Ltd (0703 452221) is distributing a software based training package for UNIX System V 4.0 and C. Running under DOS and some PC UNIXs, the software is supposed to be the equivalent of 103 hours of lecture-based training. This seems like a good idea, although one flinches a bit at the price (£999). A free demo disk is available.

Letters

We welcome short letters on any subject that is relevant to software development. Please write to The Editor, .EXE Magazine, 10 Barley Mow Passage, Chiswick, London W4 4PH. Unless your letter is marked 'Not for Publication', it will be considered for inclusion on this page.

Left-handed Graphics

Sir,

I welcome the article by Graeme Webster on 3D computer graphics. Before reading the rest of the series, I would like to make a point about the 2D representation of the 3D axes. Many times I have stared at the symmetrical lines of Figure 1 and wondered if the origin of the lines represents the far corner of a room viewed from inside, or the near corner of a cube viewed from outside. The difference is, of course, crucial.

Rather than drawing three symmetrically placed lines, I prefer to mimic the human hand as used by a photographer or artist to 'frame' a view, as in Figure 2. Here the thumbs align with the horizontal (H), and the first finger, naturally, with the vertical (V). If the second finger of either or both hands is pointed towards the viewed object (and it is difficult to point it in any other third direction) then that second finger points to 'depth' or 'distance' (D). This de-

piction can be stylised as in Figure 3.

I arbitrarily but conveniently assign the three axes to the thumb and fingers in sequence thus: x,y,z to thumb, first finger, second finger respectively.

The **left** hand fortunately then matches the rule generally applied to the Graphics screen and to the drawing board, where the origin is in the bottom **left** corner.

The architect can, if he wishes, use the right hand, turning the wrist easily so that the second finger points towards him. Then $H=x$, $V=y$, D (in this case 'height') $=z$. But why bother? We can all interpret height as a negative depth, so let's *all* use the left hand!

In short, **left** hand: thumb, 1st, 2nd = H,V,D maps to x, y, z for a normal view.

I am now ready for the rest of the series!

A left-handed DAG Tait

Dorking

Surrey

Graeme Webster's series continues this month, with an article on floating horizons.

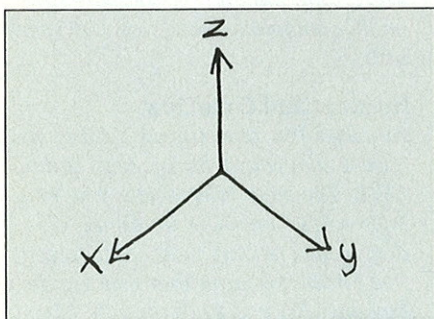


Figure 1 - Conventional 3D axes

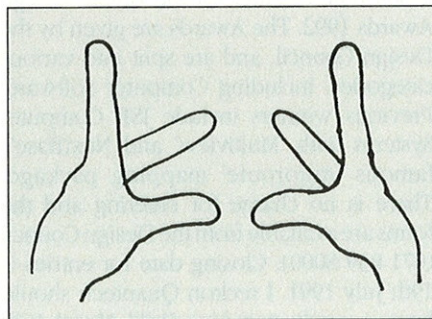


Figure 2 - 'Framing a view'

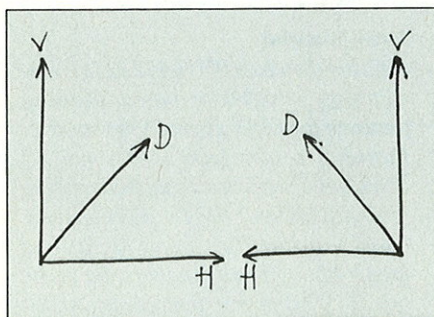


Figure 3 - Stylised 'Framing a view'

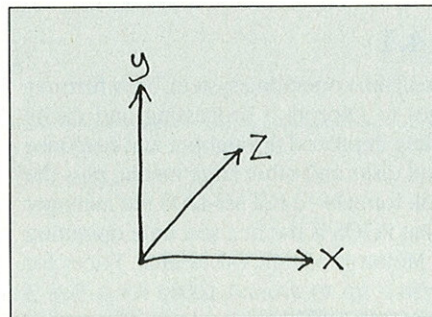


Figure 4 - A left-bander's solution

PIC whose brains?

Sir,

You may be interested in a little problem that I had attempting to program the 8259 programmable interrupt control chip on my PC. With help from several books, I was writing some simple routines that could be used within C that could access the serial ports. However, when I ran my code, it didn't work. I looked through my code, and I looked through the book examples, and I could not see anything wrong.

'Over to you, CodeView,' I said reluctantly (not being one that likes it very much) and I found what I thought was the problem. When I tried to mask off the IRQ3 bit (for COM2) on the Interrupt Mask Register (IMR), something strange was happening. The first step in this little operation is to read the IMR through the I/O port (in al, 21h), then set bit 4 to zero to enable IRQ3 (and al, f7h), and then to output the new IMR back to the controller chip (out 21h, al). But CodeView had other ideas. When I read the IMR into al, CodeView reckoned that the value was 'FF'. According to my documentation, that meant every interrupt (including the keyboard and clock) was disabled. What a load of rubbish!

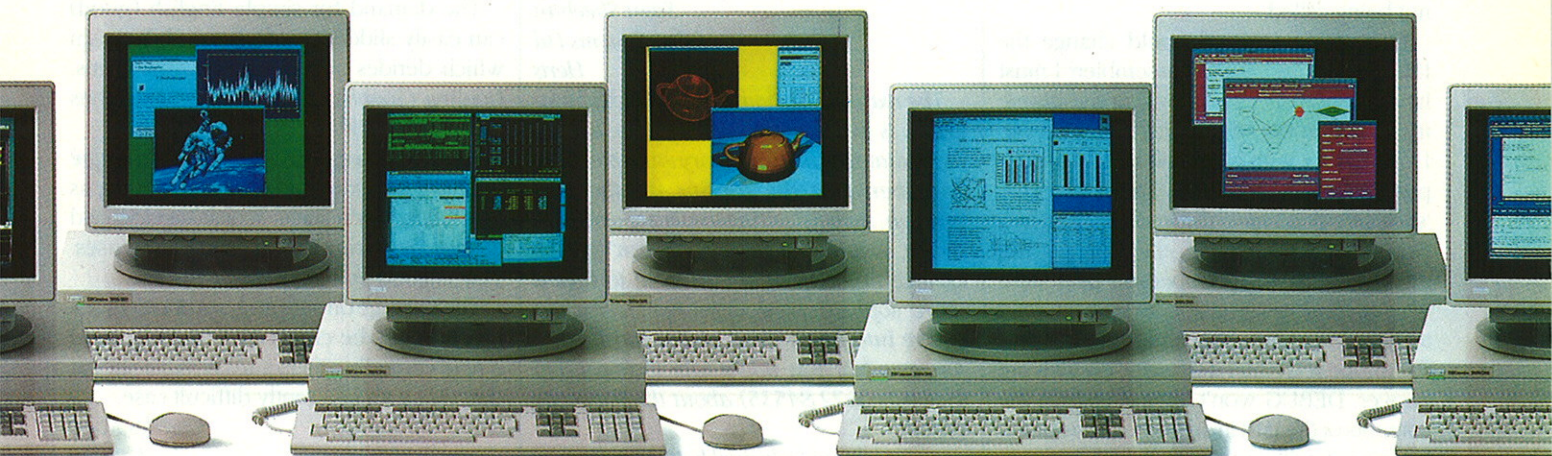
With an even bigger sigh I went over to the dreaded DEBUG. I ploughed my way through the DOS manual to see what I could do. The commands 'I' and 'O' allow direct I/O. Here is what I typed in:-

```
I21 <cr>
O21 A0 <cr>
I21 <cr>
```

The first I21 returned a value of 'A8', so I changed it with the O21 A0, writing 'A0' to the IMR. The I21 verified that it had worked, which it had. The next step was to set the mask back to 'A8' and enter some assembler code:

```
xor ax,ax ;this clears ax
in al,21 ;read in IMR
and al,f7 ;clear bit 4
out 21,al ;stuff it back
in al,21 ;and read it back
```


Workstations from Digital.



To get the most out of
your applications ...

I then traced through this, using the 'T' command. After the first `in al, 21, al` holds the value of 'A8'. The `and` operation changes `al` to 'A0'. I output `al` to the IMR register and read it back. The `al` register now holds 'A8' again. The contents of the IMR did not change, which would explain why my IRQ3 interrupt service routine was not being called.

How come DEBUG could change the IMR register and not the assembler? I must have spent nearly two days on the phone trying to find out what I was doing wrong. Don't bother with Microsoft technical support if you have a problem - they always say 'sorry, we don't have the knowledge here'. Nor IBM for that matter, they are just not interested. There were people who were very helpful, especially the guys at Intel technical support. After almost an hour on the phone with Intel they came up with the answer. DEBUG won't let you change the contents of the IMR in an assembler routine, because it will overwrite it with its own value. As for CodeView, well it just gets its knickers in a twist. So I tried what Intel suggested (run DEBUG, record the IMR value, quit DEBUG, run a program that changes the IMR, enter DEBUG again and then read the IMR). At last it worked. I had been chasing a red herring.

This leads me to the following questions. If you are developing a low-level assembler program, how can you safely debug it? And who can you talk to with technical enquiries?

Oh, and the cause of my initial problem was that my COM2 board is broken - it works fine on COM1.

Sean Sheehan
Fibre Systems Ltd
Herts

The way I debug low-level assembler programs is by inserting code to write out intermediate results to memory variables. You can then run the routine all the way through, without setting any trace or breakpoints, so that any debugger doesn't get the chance to interfere. If you need to take a more sophisticated approach, I reckon you're into fancy (and costly) hardware - for example, you might talk to Roundhill Systems (0672 84535) about the Periscope system.

As for getting technical help, you already seem to have found an oasis in the technical support desert - more fool you for telling us, as Intel's switchboard will now probably be jammed by fellow .EXE readers. You could join the CIX conferencing system (voice 081 390 8446) which allows you to pose questions to a mixture of experienced pro-

grammers, who have done it before, and busybody show-offs, who wish to impress with their knowledge - so sorting out the replies might be a problem.

P-word apologist

Sir,

The demand for simple English (good) can easily slide into a form of Philistinism which derides any use of unusual terms. Cooling (Soapbox, .EXE May '91) has thus slid in his article.

What is wrong with the word *paradigm*? We have not extended its meaning as far as the philosophers have. Words do twist and change, acquiring new meanings and uses. That is good and interesting.

The passage on subprogram declarations he quotes at the end is easy to understand if read carefully and is a reasonable shot at describing an inherently difficult case.

Charles Rowe
London

Letters submitted to this page may be edited. The writer of the best letter of the month, as judged by the Editor, will be rewarded by a T-shirt or similar-valued .EXE trinket. The best letter is the one printed first.

Graphics Server for Visual Basic

- Harness the amazing power of Graphics Server quickly and easily using our specially developed Custom Control.
- Integrate graphs and charts directly into your Visual Basic applications.
- Manipulate your graph objects using standard Visual Basic properties, events and methods.
- Present your data as pie, bar, line, scatter, area, gantt, polar and high-low-close graphs.
- Add statistical lines and trends.
- Create high-quality hardcopy.
- Interchange images with other applications via Clipboard and Metafile.
- Pay no royalties on the applications you create.

Bits Per Second Ltd
14 Regent Hill
Brighton BN1 3ED
Tel. (0273) 727119
Fax. (0273) 731925

only £99.00 + VAT
comes complete
with a royalty-free
run-time copy of
Graphics Server.

CIRCLE NO. 033

Affordable CASE Tools

Personal-SELECT at £495 *

A full function PC-based Analysis and Design Tool supporting the YOURDON (with Hatley and Ward Mellor real-time extensions) or HOOD (version 3.0) methodologies

Project-Select from £995 *

Links Personal-SELECT users working in projects and teams in a true multi-user environment

The SELECT product range has been developed in the UK by SELECT Software Tools Ltd. Great Western Instruments is able to supply full technical support including turnkey systems and training courses.

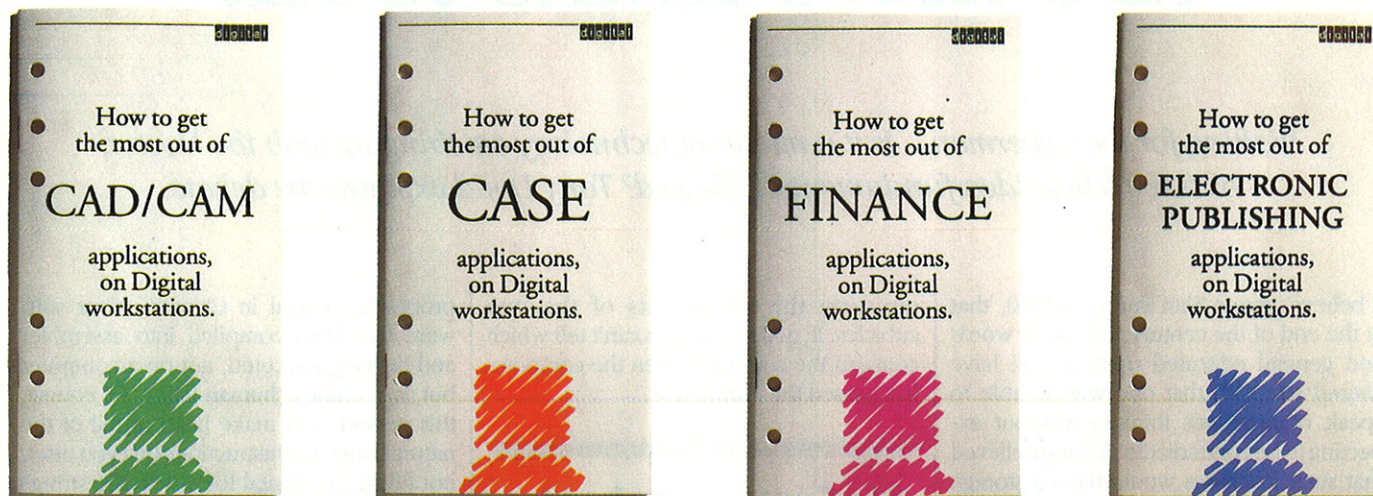
* Carriage free, excluding VAT.
Order now from:

**GREAT WESTERN INSTRUMENTS
LIMITED**

32 Paulmont Rise, Temple Cloud
BRISTOL BS18 5DZ

Tel: 0761 52116 Fax: 0761 53226

CIRCLE NO. 032



Take guidance!

You've found your application ... you've seen it running ... and you know what it can do.

Just two more decisions to make ... what platform to use, and where to get support for the smooth running of the whole solution.

Digital makes these decisions easy. We've taken four of the most popular application areas – CAD/CAM, CASE, finance, and electronic publishing – and produced a free guide for each that covers the commonly used applications packages, and the machine and network configurations that will help you get the most out of them.

Specific topics include ...

- how to get your products to market *faster* with CAD/CAM, by integrating design engineers with the whole design team ...
- Digital's COHESION environment for CASE – for software that's successful, on time, serviceable to spec., and at the right price ...
- how finance systems can integrate applications to deliver intelligence right to the desk-top ...
- how Digital provides electronic publishing solutions that go beyond the desktop, covering the complete document management and distribution process.

Each guide covers the comprehensive support Digital offers, and is accompanied by information supplied direct by third parties.

See how they run on workstations from Digital

To match your choice of applications, Digital offers a choice of workstations – with a wide range of price/performance options. Whatever your application there's a Digital workstation to run it, from entry level VMS or UNIX® through to high-performance UNIX workstations, VMS and UNIX servers – all with unmatched capability for system expansion and integration into your enterprise.

And support? From Digital, it's comprehensive – through service, training and consultancy. In fact, support from Digital starts *before* you've even chosen a Digital workstation, with these four new Workstation Application Guides.

Send for your *free* copies – today!

Free, pocket-sized, packed with information – these are guides you should have *now*. Choose one, or ask for the whole set. For your copies post the coupon today.

To: Application Guides, Digital Equipment Co. Ltd,
PO Box 525, RHA-PR, Maidenhead,
Berks., SL6 1YU.

Please send me the application guide for:

- ☒ (Tick) ☐ CASE
☐ CAD/CAM ☐ Electronic
☐ Finance ☐ Publishing

☐ Details of Digital's
workstations range

I intend buying workstations in the
next 6 months? ☐ Yes ☐ No

Name (Mr, Mrs, Ms) _____

Position _____

Organisation _____

Address _____

Postcode _____ Tel _____

Industry _____

EXE 7/91

Are there limits to AI?

Waiting for the Cybermen... Is it a matter of technology catching up with the Doctor, or is the whole idea fundamentally flawed? Tony Dodd explains the debate.

'I believe', wrote Alan Turing in 1950, 'that at the end of the century the use of words and general educated opinion will have altered so much that one will be able to speak of machines thinking without expecting to be contradicted'. Turing believed that such machines would have a storage capacity of about 10^9 , not at all a bad guess; but in other respects it would be hard to be as optimistic as Turing about progress in the next nine years.

Turing's view was that programs would be written that would settle the question of machine intelligence irrespective of philosophical objections; in particular, a program that could play the imitation game, so that an interrogator could not tell the computer from a human respondent. Depressingly for supporters of AI, such achievements still appear to be at least 50 years away. But the arguments intended to show that machine intelligence is impossible have been voiced with increasing volume in recent years, most recently in Roger Penrose's book *The Emperor's New Mind*.

The Turing test

The factions can be divided into two groups:

- those who argue that it makes no sense to say that software is intelligent;
- those who argue that though it makes perfectly good sense to speak of intelligent software, in practice such software can never be written.

Just how intelligent is intelligent? A test often proposed is the Turing test, alluded to briefly earlier on. You have two cubicles, one containing a human being and one a computer fitted with voice understanding and speech generation contraptions. You cannot see into the compartments. You have five minutes in which time you can

question the inhabitants of the two cubicles. If, at the end, you can't tell which contains the computer, then the computer has passed the Turing test.

Unfortunately the only conclusion we can reach is that we were sold a pup

We may divide the two groups of opponents of AI according to their views on the Turing test. The first group holds that even if the test were passed we could not call a machine intelligent; the second, that the test can never be passed.

Chinese rooms

The first group of opponents of machine intelligence argue that intelligence is different from mechanical computation, which is all that computers can do. Their best known representative is Searle.

Searle's argument, often called the Chinese room argument, can be summarised as follows. Suppose you could write a piece of software so intelligent that nobody could tell it from a human being. In one room you have a human being and in another a PC running your software; but without looking inside nobody can tell which room. Searle denies even so that the PC exhibits understanding. Suppose, on opening the door to the room that was supposed to contain a PC we find instead somebody who speaks Chinese but no English, a mountain of assembler code and a copy of the Intel 80386

processor manual in Chinese. Your software has been compiled into assembler and is being executed, not on a computer but by a Chinese human being. Of course, this person can't make head or tail of the natural language instructions that you utter, but filling his mental RAM with the strings he hears he resolutely follows the assembler code until he reaches an answer, which he enunciates. Now, Searle says, it is nonsense to ascribe understanding of the dialogue to the Chinese emulator; he no more understands it than the Hungarian whose hovercraft was full of eels understood English dialogue.

The standard reply by supporters of AI is the claim that it is the totality Chinese plus code plus manual that exhibits understanding, not any one component; and if anyone suggests that it is odd to ascribe understanding to so peculiar a collection of objects the rejoinder would be that it is also odd to ascribe understanding to a joint of meat pervaded by tiny electrical currents.

Certainly the tendency to ascribe intelligence to the routine operations of machines is overdone. It is said that one of the by-products (according to the malicious, the only by-product) of the Japanese 5th generation project is an intelligent toaster; so now I must regard my ordinary domestic toaster as unintelligent, which seems unfair to it. Searle would come to its rescue by saying that it was not unintelligent, but rather the sort of object to which it is meaningless to ascribe intelligence. There are AI authorities who would ascribe beliefs to thermostats, surely a strange misuse of language. Intelligence must always involve some ability to reason about the activity performed and justify it; thermostats cannot do this (I hesitate to speculate about the toaster). By shooting down some overblown claims for early AI systems, Searle gives the impression that he is demolishing the entire field.



What computers can't do

The second group of opponents of machine intelligence would be willing to concede that a computer that passed Turing's test was intelligent, but they deny that any computer can ever do that. To this end they display a function that no computer can ever calculate but that a human being can.

It may seem odd to try to outdo computers on their home ground in this way. Surely the most obvious evidence that computers aren't human is that they don't fall in love, write poems, behave irrationally and so forth. Computing a function is one thing that we do expect a computer to do; and if tomorrow anthropological research were to show that in fact people spend their time evaluating integer functions rather than falling in love and all the rest of it, most of us would probably concede that AI was on the way to success.

Again, how on earth can we ever prove that no computer could ever compute a function? Maybe my PC can't manage it, but have you tried it on a Cray? If it can't be programmed in C, why not C++? And if all else fails, who knows what next year will bring?

The answer to the first question is that we have a precise mathematical theory of which functions computers can calculate, whereas our views of the limitations of computers as poets and dreamers tends to be based merely on personal experience.

To say 'computers don't fall in love' will always be to invite debate, but to say 'computers can't compute the set $\{e: \{e \leq t\} / (e) \text{ converges}\}$ ' admittedly a statement less likely to break the ice at parties, will save you a lot of controversy.

The answer to the second question is quite extraordinary: it is generally believed that all computers compute exactly the same functions. Of course some take longer than others, use more space and so forth, but in

```
/* This program loops with input n
if and only if program n
with input n doesn't loop */
int main(int n)
{
    if (!loops(n,n))
        g: goto g;
    else
        exit;
}
```

Figure 1 - LOOPKILLER!
test program

the long run, the PC on your desk, the Cray and anything at all that may come along next year are all equivalent. They are all equivalent to a number of ideal computing devices invented before real computers had been built, the best known of which is called a Turing machine.

By shooting down some early AI systems, Searle gives the impression that he is demolishing the entire field

Before you tip the PC out of the window and rush out to buy a Turing machine (and there used to be several Turing machines on the market. The one I used was part of an Open University course on logic, and implemented the full Turing spec except that instead of having an infinite number of cells the tape had, as I recall, eight. But the user could install extra memory with a pair of scissors, a pile of scrap paper and some sticky tape.) with tape attachment and the latest release of TMTOS (pronounced Tomatoes) pause a moment; these ideal equivalences cost a lot in real computation time.

In real programs, the question is not 'what can we do in principle' but 'what can we do with the resources available'. However, the result is still extraordinary: differences between computers are differences of speed rather than in what can ultimately be computed. We can never prove this result, and it depends for its persuasiveness on the enormous number of different systems that have all been shown to be equivalent; so we speak not of a theorem but of a thesis: Church's thesis.

If we accept Church's thesis, then we only have to display a function that could not be computed on a PC using C and we have a function that cannot be computed by any computer anywhere until the end of time. In fact it is quite easy to find a function. Set up your PC, and consider the programs you might write for it. Once compiled, any program will be a sequence of bytes in memory, and could in principle be regarded as a colossal integer. Programs demand input as well, but for simplicity, assume that each

program just reads one argument on input. So, for example, the program that is integer N in memory might find that its input is M . It will execute, and though many weird and wonderful things may happen as a result, all that interests us is that it may stop (by reaching the compiled form of an exit statement) or it may not.

You might think that this question is of such practical interest that programs must exist to detect non-terminating code. You might even succumb to the advertisement for the revolutionary loop-checking program LOOPKILLER! release 0.98b. This tells just by examining your code whether or not it will terminate. You may assume that it looks at the code, detects, for example, instructions that jump to themselves, and decides whether or not you are heading for a loop. You rush off and buy it.

When you get the program you find that it's even better than the advert suggested (remember this is a purely fictional account). Not only can the program run stand-alone, it can be incorporated into other programs. One day, somebody takes the package off the shelf and they write another program as follows:

Given an input integer n , ask LOOPKILLER! whether the program that compiles to integer n terminates when its argument is n . If it does then go into a fatuous loop, otherwise stop at once. The C code is given in Figure 1.

Compile this program and, of course, it too has a number, which we can call m . Just suppose its number was 5016673245. If we ran it with input 23 would it terminate? Well, it would terminate if program 23 with input 23 *didn't* terminate. How about if we ran it with 5016673245? By the same argument, it would terminate if and only if program 5016673245 with input 5016673245 didn't terminate. But it's nonsense to say 'it will terminate if and only if it doesn't terminate!' Either it terminates or it doesn't.

To argue more precisely:
Suppose that when executed with input m , program m terminates. In other words $! \text{loops}(m, m)$ is true. But then program m with input m enters that nasty little loop and certainly doesn't terminate. Oops! So -

Suppose that when executed with input m , program m loops. In other words $\text{loops}(m, m)$ is true. But then program m with input m terminates via that nice exit statement.

Unfortunately the only conclusion we can reach is that we were sold a pup; there

could not possibly be such a general loop checker as LOOPKILLER! For it claims to compute a function

```
BOOL loops(int, int);
```

that no computer could ever compute.

Now suppose that the human brain could compute this function: then we should have shown that no computer can ever be the equal of the brain.

What we know about numbers

Unfortunately this nice precise argument now tends to wander off in a more philosophical direction. Before we wander off after it, we might at least notice that no human being has yet come forward offering to compute this function; the claim that the brain can, in principle, compute it is founded on a somewhat circuitous argument.

Why should anyone think that they could compute `loops(n, m)`? `loops(n, m)` is just a statement about numbers, as is $n+m=p$, so we might start by asking how we know, for example, that $2+2=4$.

We need to go a little further back and ask: how can we tell that anything at all about numbers is true? Some things we can, perhaps, tell by experience, $2+2=4$, for example, but how can we ever test the truth of the proposition that every number is either odd or even? Certainly not by checking them all! Yet in fact it's very easy to prove this: given a number, take its remainder on division by 2. If this is 0 it's even, if 1 odd. Here we supposed that every number n can be expressed $2m+r$ where $r < 2$, which itself needs proof. Ultimately we can chase the truth back to a few things that we have to accept as obviously but unprovably true; these are called *axioms*.

Now we must derive another technical result. Again it tells us something about what computers can and can't do. I want to start by asserting two things that I haven't space to prove. The first is that it is possible to represent a proof as a number, probably a

```

BOOL loops(int n, int m)
{
    int p;
    for (p=0; p++; )
        if (proof(p, ls(n, m)))
            return TRUE;
    else
        if (proof(p, neg(ls(n, m))))
            return FALSE;
}

```

Figure 2 - `loops()` implemented using `proof()`

stupendously big one. Just as any program was viewed as an integer, so any proof could in theory be stored in a computer's memory as an array of character strings, and this could be viewed, using a suitably idiotic union declaration, as an integer. Some integers may represent sequences of character strings that are not proofs; all that we require is that all the proofs are there, however thinly scattered.

How on earth can we ever prove that no computer could ever compute a function?

The second thing to say is that there is a program `proof(p, t)` that says that p is a proof of t . `proof(p, t)` always terminates when it is run and returns TRUE if p is a proof of t and FALSE if it isn't. To persuade you of this I could urge you to consider that it is easy to check that something is a proof, but hard to find a proof. You should note, however, that the computer program can only carry out this assignment if the set of axioms is such that the computer can recognise whether or not a character string is an axiom or not. Of course, if the set of axioms is finite this will not be a problem; you'll see later why I harp on this point.

Now suppose that the set of axioms is such that every statement about the numbers is either proved or refuted by them. In particular note that `loops(n, m)` is just a statement about numbers. Moreover, using simple character manipulation it is easy to see that there is a function

```
char* ls(int, int);
```

such that `ls(n, m)` is the string "`loops(n, m)`" with values for n and m suitably replaced.

Now consider the program in Figure 2. The problem is that we saw that this function can never be written! So one of the assumptions that we made is wrong: either there is some statement about numbers that is not settled by the axioms, or the axioms are so complicated that a computer cannot even recognise whether a string is an axiom or not. Put in more formal terminology, if the set of axioms is

recursive then it is incomplete. This discovery, made by Gödel in 1931, came as a tremendous blow to mathematicians who thought that the whole of maths could be derived from a few simple axioms.

Plato and Penrose

The unprovable proposition of Gödel's work was quite far fetched, but in fact there are some quite simple problems in mathematics that cannot currently be proved or disproved from the usual axioms.

Here is an exercise: very easy to program, very profitable for you if you make it work. Every schoolchild knows that the square on the hypotenuse of a right angled triangle is equal to the sum of the squares on the other two sides. Moreover, there are lots of sets of whole numbers that can satisfy this relationship: for example

$$4^2 + 3^2 = 5^2$$

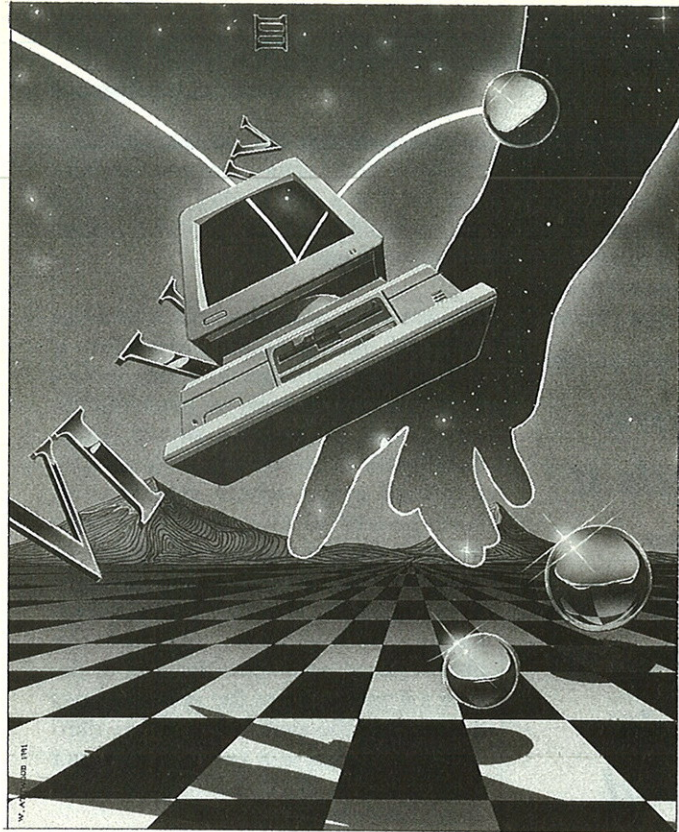
But can you find positive integers p, q and r and some n greater than 2 where

$$p^n = q^n + r^n$$

Before you get going, you may care to know that it has been shown that n must be at least 253747889; this may affect your choice of C compiler for the problem.

Now on the one hand, nobody has ever found such numbers; and on the other hand, nobody has ever proved that such numbers cannot exist. The tiresome thing is that Fermat wrote in the margin of a book the claim that he had such a proof but that it was too long to fit in the margin; and then died before he could write it out. With a lesser mathematician nobody would hesitate to accuse him of wishful thinking, but given Fermat's other achievements, the suspicion that he may actually have had a proof will not go away. What would we say if it turned out that the proposition could be neither proved nor refuted? (In fact this is quite unlikely; but there have recently been several quite natural combinatorial principles discovered that can neither be proved nor refuted from the usual axioms.)

Philosophers of mathematics differ about such statements. Some (the formalists) would say that mathematical truth is solely about provability from axioms. This statement, then, is neither true nor false, though there will be formal models of the axioms where it's true and others where



ADD NEW DIMENSIONS TO YOUR SOFTWARE

... by bringing your packaging into perspective.

Matchpoint Design pinpoints marketing solutions for software

packaging that stands out from the crowd.

Our fresh ideas in imaginative design combined with the latest in

artwork technology gives your software programs the competitive

edge.

From a simple line design to powerful, full colour illustrations

and corporate image we provide a one-stop service for all your

packaging needs whether large or small.

To find out how we can give your project a new perspective

contact: Peter Jones on 021-702 2159.

MATCHPOINT

D e s i g n

Centre Court, 1301 Stratford Road, Hall Green, Birmingham B28 9AP.
Telephone: 021-702 2159 Fax: 021-702 2085.

CIRCLE NO. 966

C-scape™ 3.2

The latest release of C-scape 3 — the best selling character/graphics based screen interface C library in the UK. Runtime royalty free. Includes Look and Feel Screen Designer™ and Source Code. DOS, OS/2 £370
Also available for Windows, UNIX and VMS.

Heron C-scape Software Extension Library



Enhanced set of functions for C-scape 3 Price
Includes Source Code..... DOS £99.95
Unix £199.95

Phar Lap 386 | DOS Extender™



Beat the 640K DOS limit with a flat 32-bit address space on 386 processors. Voted best of 1989. Supports a wide range of 32-bit languages.SDK £315

NEW Phar Lap 286 | DOS Extender™

£315

db_VISTA™ III — Release 3.2

Complete family of C library dbms functions and database management utilities for DOS, Windows, OS/2, Macintosh, UNIX, ONX, VMS. Source code available — runtime royalty free.

Single-user DOS, OS/2 or Windows DBMS object libraries..... £435

Ask for details of other options and multi-user pricing.

MagnaCharter II

Enhanced version of the much acclaimed MagnaCharter charting package for Windows 3, 99x99 cells with wide range of printer support. Complete set of symbols — or add your own..... £160

MagnaCharter

Still available, MagnaCharter for DOS (includes runtime Windows 2.)..... £118

PERSONAL ACCOUNTANT

The only complete personal finance package including — Accounts, Assets and Liabilities and Inventory and Insurance.....DOS £149.95
New Tax Reckoner module £119.95



All Systemstar products are distributed and supported by Systemstar. Prices are exclusive of VAT and subject to change. Orders and enquires to:

SYSTEMSTAR

L I M I T E D

1-3 Parliament Square, Hertford, SG14 1EX
Telephone: (0992) 500919 Facsimile: (0992) 554261

CIRCLE NO. 967

it's false. Others (the Platonists) would say, yes, it is true or false. After all we happily judged the axioms to be true, so we must have some perception of truth in mathematics. Followers of Plato believed that the world of numbers was more rather than less real than the perceived word, and they believed that we could judge the truths that hold in that world through reason. If that is true, and we can, through some kind of internal facility in the brain, judge the truths of mathematical propositions, then we can evaluate a function that no computer can ever evaluate; this inner mathematical facility will forever defy the efforts of AI researchers.

Naturally there is some debate about the reality of this claimed source of mathematical knowledge. Few mathematicians are formalists just as few theologians are atheists; the subject is more interesting if its object of study exists. Claims of Platonic insight have to be hedged about to a great extent; no single mathematician will come forward offering a yes/no answer to any question of mathematics whatsoever but, it is claimed, as mathematics makes progress the human race discovers more and more about the Platonic realm, in such a way that eventually any proposition can be determined to be true or false; and no computer can do this. A formalist will reply that indeed as time goes by axioms may be added but that this is not a question of the discovery of new truths but simply an arbitrary decision to add a new axiom whose negation we might just as well have added.

Roger Penrose is a champion of the Platonist standpoint; in his book *The Emperor's New Mind* he argues the case against AI and suggests that a new physics of the brain may explain its access to non-computable functions (if you read Penrose, you will find that I have simplified his argument a lot: he does not need the full Platonist assumption, merely a claim that we can judge the consistency of a certain theory, which he constructs in a way similar to Gödel's original proof).

Before concluding, we should return to the Turing test. When the cubicles are locked, what will Penrose's question be? It needs to be a sentence that is true of the natural numbers but does not follow from the usual axioms. The computer, so the theory goes, will flounder, trying both proofs and refutations until its time is up, the human will gaze into the Platonic realm and return with the answer. Naturally, we shan't be able to prove that the human answer was the

right one, but after all the computer didn't answer at all, so there was no trouble telling them apart. But maybe a computer programmer with insufficient respect for the mathematical profession will install a routine that tries for (say) four minutes to prove or refute the sentence, and then guesses at

He no more understands it than the Hungarian whose hovercraft was full of eels understood English

random, but adds the guess to the axioms to make sure it doesn't answer the next question inconsistently. Now what can we do? Of course, nobody would suggest for a moment that there's anything in common between the intuition of a brilliant mathematician (who we happen to have locked away in the cubicle) and guesswork, but they are embarrassingly difficult to tell apart.

Conclusion

It would be fair to say that workers in AI do not lose a great deal of sleep about these results. There is still so much that can be done and needs to be done that the idea of reaching a theoretical boundary is not of practical relevance; limited research funding imposes a much more painful constraint.

But how do these debates about AI concern those of us who write what we call AI software? Presumably we all believe that the development of computer systems will be measured largely by the extent to which systems are produced that are more accessible to ordinary human understanding, and require less understanding of special formalisms. If you follow Searle and Penrose and deny the computer access to human understanding then you will probably subscribe to the view that although individual AI techniques are useful, AI has no co-ordinated part to play in the development of software; we shall add these techniques to our toolkits and use them or not as circumstances demand. If, on the other

hand, you believe that AI can provide a computer model of the brain, you will probably see it as a new kind of software that will eventually replace the old machine-based paradigms.

My own view, as perhaps the reader has gathered, is that Searle's argument is without practical interest, in that it does not restrict the systems we can build but merely tells us how we should badge them; while Penrose's argument rests on a claimed insight whose existence can never be verified. Of course, to disagree with two particular arguments against AI need not lead us to argue that all arguments against AI are false. The best arguments, in my view, are not the philosophical arguments that aim to deliver a knock-out but the practical ones that say that the aims of AI are very difficult and that present research does not suggest that machine intelligence is just around the corner.

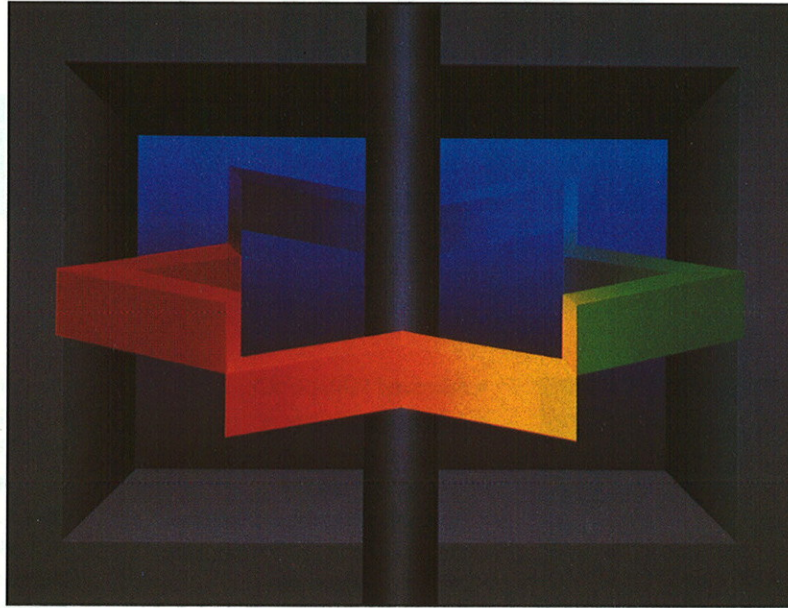
On the positive side, I believe that there is plenty of evidence that machines can be programmed to display intelligence. I am not talking about the production of software so skilfully crafted as to be indistinguishable from the brain; nobody can tell whether that is possible or not. Supporters of AI do not necessarily believe that every last detail of the brain can be mapped in software, but we do believe that the brain and the computer share a formalism - logic - and that although the physical details of implementation differ wildly from the brain to the computer, logic provides a level of abstraction at which the two are comparable. This is not one of those broad generalisations that prove hard to refute because they move around so easily; it is a tentative view of the brain that will be confirmed or refuted by the development of computers.

The questions raised by AI are among the most important that face us as we struggle to come to terms with the computer, and yet in a sense they have their roots in problems that worried people long before there were computers. Though the philosophical discussions about AI are interesting, the decisive answer depends on the research and programming that will be done in the decades ahead.

EXE

Tony Dodd is the Chairman and Managing Director of Expert Systems Ltd. He was the designer of the Prolog-2 implementation of Prolog, and has written textbooks on that language. He can be contacted on 0865 794474, or emailed at tdo@uk.ac.exeter.cs.

Who put C++ to Work?



glockenspiel
class constructors

Glockenspiel C++

Glockenspiel C++ leads the field in object-oriented programming. It's the most effective and efficient implementation around. It conforms exactly to the AT&T C++ specification – and they don't come any closer than that! Portable too: it's on more platforms than any other C++ implementation.

Glockenspiel CommonView®

Glockenspiel CommonView is the leading C++ class library for developing Windows, PM and OSF/Motif applications. Logical, well structured and easy to use, CommonView is 3 to 5 times more productive than the API.

C++ Training & Consultancy

Glockenspiel C++ training courses reflect our experience as C++ developers. From design to development, *learn from the best!*

Our C++ consultancy services match your needs as you commit more resources to C++ and object-oriented development.

Further Information

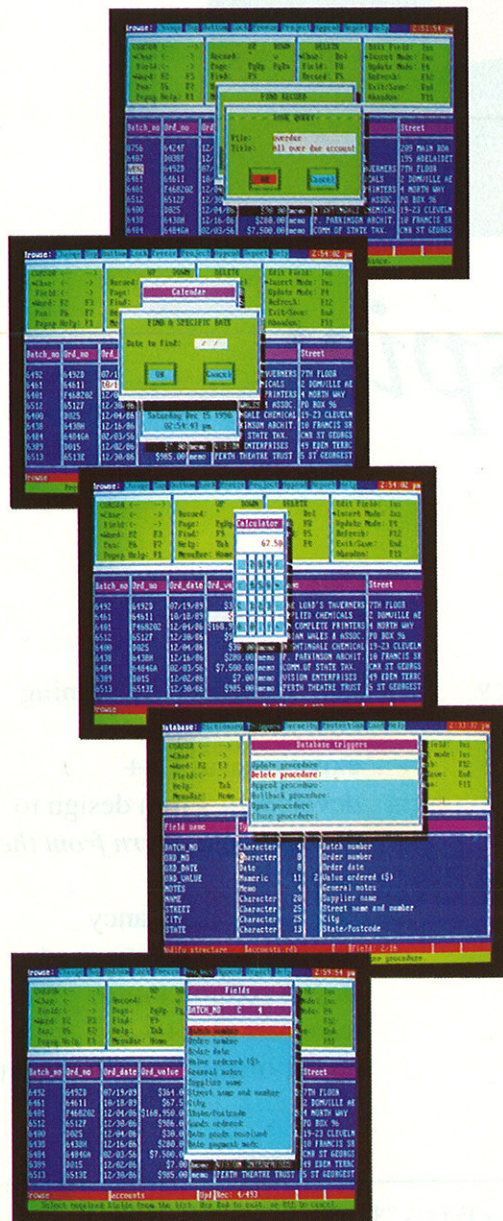
GLOCKENSPIEL, 39 Lower Dominick Street, Dublin 1, Ireland. +353 (1) 733166. Fax +353 (1) 733034

■ **US:** IMAGESOFT, (516) 767-2233. Fax (516) 767-9067. ■ **OASYS**, (617) 862-2002. Fax (617) 863-2633 ■ **UK:** QA TRAINING LTD., (0285) 655888. Fax (0285) 650537.
■ **Italy:** INFERENTIA, (02) 26680568. Fax (02) 2364258. ■ **Sweden:** LINSOFT, (01) 3124780. Fax (01) 3152429. ■ **Germany, Switzerland, Austria:** PSI, (06021) 492-0. Fax (06021) 492-112
■ **Benelux:** RIJNHAAVE, +31 (71) 218121. Fax +31 (71) 216118

Glockenspiel CommonView and Glockenspiel C++ are registered trademarks of Glockenspiel Ltd. The trademarks of their respective corporations are acknowledged.
Glockenspiel "Colour Rotation" logo by Francis Tansey

CIRCLE NO. 968

Recital 7.0 Is The Only Choice.



- Advanced RDBMS/4GL
- Active Data Dictionary
- Object-oriented with pre-defined "application objects"
- Open client/server architecture
- Integrated pop-ups for calendar, calculator, choice lists
- Built in "Dynamic Compiler" New v7.0
- Pop-up time management scheduler New v7.0
- Relational multi-record table fields in forms New v7.0
- Split screen relational browse New v7.0
- Database, form and report triggers New v7.0
- Automatic record locking
- Supports 14 datatypes including memo, numeric, byte, integer, short, character, date, logical, float, real, packed, zoned, vaxdate, quadword

- Available now on 100+ VAX/VMS and UNIX computers (including full VAX cluster support)
- SQL (interactive and embedded)
- dBASE, FoxBase and Clipper compatibility (yes, even on VMS!)
- Terminal (including full color support) and keyboard independence
- Transparent connectivity to existing RMS and RDB databases on VAX/VMS
- Completely integrated development environment
- Powerful end-user tools

Recital
CORPORATION

Rdb
RSVP
SOLUTIONS • VENDOR • PROGRAM



Recital Corporation LTD, South Bank Technopark, 90 London Road, London SE1 6LN, United Kingdom, Tel 071-401-2727, Fax 071-633-9617
Recital Corporation Inc., 85 Constitution Lane, Danvers, MA 01923, Tel (508) 750-1066, Fax (508) 750-8097

CIRCLE NO. 969

All constraint is NOT evil...

Ever tried to schedule a school timetable - or plan the layout of a magazine? Bet you thought, 'Here's a job for a program'. Al Roth and Stuart Watt explain how it is done.

Languages such as LISP and Prolog have traditionally been regarded solely as AI programming languages. We have argued in the pages of this Magazine (and elsewhere) that they are powerful vehicles for the rapid deployment of a wide range of real-world commercial applications. Both languages will always have a niche for a specific set of programming tasks, despite the powerful C++ juggernaut that seems to be rolling over us now. Their place is assured, because languages like LISP and Prolog are much more geared to solving certain kinds of problems. One such domain is scheduling.

In this month's *Soapbox*, Al suggested that many ideas have migrated from AI into conventional Computer Science. The technique of *constraint programming*, once firmly within the domain of AI, is now coming into widespread use in a range of application areas. Several constraint languages have been released commercially recently, and PROLOG-3, developed by Alain Colmerauer at the University of Marseille (the creator of the first ever Prolog program), adds the notion of constraints to standard Prolog. In the LISP programming community, constraints have been around for many years.

In this article we are going to look at the general problem of scheduling, and attempt to show how constraint programming can assist in scheduling tasks through reducing the size of the search space. This makes it possible to quickly solve problems that otherwise could take weeks using a blind, exhaustive search strategy.

An Application

Take an example from the real world (and one incidentally that has real commercial solutions in the market-place). Imagine we

wish to build a system that schedules a school timetable. The application should take as input a set of available resources concerned with subjects, class-rooms, tea-

LISP is an intrinsically better language for building a constraint based scheduling system

chers, number of pupils etc. The output should be a consistent timetable which assigns pupils to a particular teacher, in a certain class-room, in order to receive tuition in an appropriate topic.

An additional requirement is that the system should also generate a timetable which takes into account a set of constraints provided by the user, eg

- 'Never put French next to German'.
- 'Don't put Physics, Chemistry and Maths next to each other'.

The system should be capable of dynamically propagating the result of this constraint throughout the schedule, to generate new and alternative timetable solutions. This mechanism also offers a flexible way to provide 'what-if' facilities by making it possible to add a new constraint: 'Teacher A is unable to teach maths in week 7' or 'Triple Maths is a bad idea'. The effect then

'ripples' through the timetable, eventually arriving at a solution that satisfies all of the given constraints.

Constraints are not a new idea. Abelson and Sussman (*reference provided at the end of the article*) pointed out that 'Computer programs are traditionally organised in terms of one-directional computations, which perform operations on pre-specified arguments to produce desired output'. They note that in reality we often have to model systems which are best thought of as relationships between quantities. Equations such as the formula for translating between Fahrenheit and Celsius are not one-directional; given values for either of these variables, we can work out the other. Indeed they proposed a system for local propagation in which the consequences of one calculation 'ripple' throughout the constraint network.

This is a powerful technique. It is also an entirely different style of programming to that traditionally undertaken in a conventional procedural programming language such as Pascal. Indeed, the non-procedural nature of constraint languages often makes it possible to talk about what a good solution looks like, rather than coding up an explicit algorithm for solving a problem. It is also a technique that we will attempt to demonstrate here.

Scheduling and Planning

When people refer to scheduling, they often also include planning as a closely related topic. Scheduling is related to planning, but tends to be more concerned with resources and time. Whereas planning attempts to construct a set of actions which will transform a starting state into a finishing one, scheduling is usually concerned with sequencing tasks and ensuring that the



tasks do not clash by requiring access to a limited resource at the same time.

The 'dining philosophers' problem is a scheduling problem (attributed to Dijkstra). Imagine a round dining table on which stands a single bowl of spaghetti and five forks. The problem is to write the control mechanism for five hungry philosophers in such a way that they can all eat, ensuring that 'deadlock' does not occur. Deadlock arises if all the philosophers go to the table simultaneously, pick up their left fork and reach for the right fork. The right fork has already been picked up by the adjacent philosopher. In the timetabling example, this is analogous to saying both 'Lesson A must come after lesson B' and 'Lesson B must come after Lesson A'.

Thus while the dining philosophers problem is normally associated with operating system design, it is actually a good example of a scheduling problem. Getting stuck in a scheduling problem is directly equivalent to deadlock within an operating system context. Incidentally, one solution to the dining philosophers problem is to encode the heuristic 'If it isn't possible to take both forks, then don't grab either'.

Scheduling is a member of the class of problems known as np (for n-polynomial) complete. These problems, when they are big, are usually computationally intractable. When scheduling 100 tasks, for instance, there are 4.7×10^{159} different ways the tasks can be ordered, taking into account that some might happen at the same time. In LISP, this is known as a 'bignum'! There is no point attempting to try all these

in turn, so instead we look for ways of *not* trying out all the sequences. That is, of reducing the size of the search space.

Normally, however, there is another thread to the problem - that of resources. Scheduling usually involves allocating tasks in situations in which there are a limited number

People are not very good at scheduling problems...but it is still worth looking at the way they solve them

of resources. This generally tends to constrain when tasks can happen. If two tasks require the same resources, and the resources cannot be shared (which is usually the case - for example, the resource is a person) then the tasks cannot happen at the same time.

The 'AI' line of attack on this problem is to watch the way people handle scheduling. Unfortunately, people are not very good at

it, even for fairly small problems. With big problems, such as timetabling (which may have thousands of tasks) it is inevitable that the final schedule will have faults in it. However, it is still worth looking at the way people solve this kind of problem, because even though they cannot attempt to investigate all possibilities, they can often achieve a fairly good solution. People do this by applying rules of thumb, or heuristics, which guide more directly to a solution. Heuristics come in many different kinds, some are statistical, some based on previous experience, and some based on the problem itself. A common strategy applied by people is to plan hierarchically in *blocks*. These blocks can be fitted together to make bigger blocks, until eventually the problem is solved. Heuristics are a vital part of solving scheduling problems, because they can make the search for a solution much more direct, without needing to follow so many blind paths.

The statistical heuristics have the advantage of (generally) being independent of the problem they are trying to solve. We shall look at one of these, and show how it can be used in solving a scheduling problem. When resources are limited, one task can preclude others from happening. It is the interaction between such tasks that is the key to this technique.

The tasks are all linked by constraints, so that certain variables have values which are limited by other values. When one value is changed, or its set of possible values is changed, the effects flow through all the constraints and update any other related variables. This is a very powerful technique, which can be seen in a limited extent in programs such as the Macintosh program Stella. (This shows the constraint network graphically and allows the user to change formulae and see the resulting changes to values as the effects propagate throughout the network.)

The Basic Algorithm

During the scheduling process, the basic approach is to first choose the task which is the least likely to survive any other tasks being scheduled first. For this task, choose the resources which have the lowest cost associated with them, as they are the least likely to cause other tasks to become unschedulable. When this has been done, 'commit' this task with the chosen resources. This in turn will affect the possible placing for the other tasks, which will change the order of tasks in the task queue. This is a cyclic process, which proceeds until all the tasks have been placed. If there is a problem, then the process needs to

```
;;; Main Cycle of the Scheduler

(defun schedule ()
  (loop

    ;; When there are no more tasks, then stop.
    (when (endp *task-queue*)
      (return-from schedule))

    ;; Sort the task queue so the most critical task is first.
    (setq *task-queue* (sort *task-queue* #'> :key #'task-criticality))

    ;; Choose the most critical task, and get all the solutions for it.
    (let ((task (first *task-queue*)))
      (let ((solutions (find-task-solutions task)))

        ;; Sort the solutions for the current task so the
        ;; least-impact solution is first in the set. Then we choose
        ;; that for the task.
        (setq solutions (sort solutions #'< :key #'solution-impact))
        (choose-task task (first solutions)))

        ;; We have finished with the current task, so remove it
        ;; from the queue and carry on with the loop.
        (setq *task-queue* (rest *task-queue*))))))
```

Figure 1 - Main Cycle of the Scheduler

RECESSION SPECIAL!

PRE-SALE
CONSULTING

TAKE 15% OFF ALL PRICES!

POST SALE
SUPPORT

NEW & NOTEWORTHY

Greenleaf Comm ++	£ 150
True C++ Class Library. Device independence "through inheritance. baud rates upto 115,200."	
Gupta SQL Windows for Btrieve v2.0	£ 1378
LANtastic 2mbps Starter Kit	£ 334
Economic alternative to Novell. Links as few as two PCs and grows upto 100s.	
MS-DOS v5.0 upgrade	£ 75
"Loads into extended memory. Network drivers, "TSRs in upper memory. Full screen editor, Undelete"	
Unformat and more!	
NE-1000 Ethernet Kit	£ 97
8-bit high performance network adapter.	
NE-2000 Ethernet Kit	£ 119
16-bit high performance network adapter.	
Novell NetWare v2.2	£ 599
"Combines Network 2.x, SFT, Advanced and ELS I & ELS II into one product."	
Novell NetWare v3.11	£ 2340
Complete re-write. Take full advantage of the 386 chip.	
Oracle Tools & Database v6.0	£ 674
Paradox Engine (C version)	£ 88
PC Tools v7.0 New Version!	£ 103
Turbo Pascal for Windows	£ 159
"Create, edit, compile & run programs," all from within Windows.	

ARTIFICIAL INTELLIGENCE

Arity Combination for Windows	£ 839
PDC Prolog (Turbo Prolog)	£ 202
PROLOG-2 Professional	£ 580

ASSEMBLERS & LINKERS

RTLLink	£ 191
Blinker	£ 202
Dis/Doc Professional	£ 189
Sourcer	£ 105
Spontaneous Assembly	£ 161

BASIC

Mach 2 for MS QuickBASIC	£ 55
Object Script QuickBASIC	£ 105
PowerBASIC	£ 71
Stay-Res Plus 3.0	£ 70

BUSINESS

3-D Charts To Go/Windows	£ 54
ABC Flowchart	£ 178
Adobe Illustrator for Windows	£ 273
Crickit Graph for Windows	£ 78
Grammatik IV	£ 55
Lotus 1-2-3/G	£ 479
ORG PLUS Advanced	£ 60
PerFORM Designer & Filler	£ 89
Quattro Pro Competitive Upgrade	£ 89
Show Partner F/X	£ 226
Storyboard Live	£ 278
Xerox FormBase /Windows	£ 299

C

Aztec C Cross Development Systems	£ 378
Borland C++	£ 321
C ASYNCH MANAGER	£ 123
C Communications Toolkit	£ 130
C-terp 386 with Source	£ 193
c-tree Plus	£ 386
CLEAR+ for C	£ 121
Control Pak/Windows	£ 386
DESQview API C Library	£ 110
Greenleaf Comm++	£ 110
Greenleaf CommLib Level 2	£ 252
Informix C-ISAM	£ 148
Intel 386/486 C Code Builder Kit	£ 413
Lattice C Development Sys	£ 143
Object Script C	£ 351
PC-Lint	£ 76
Pro-C with Workbench	£ 561
Turbo C TOOLS	£ 97
WATCOM C/386 Professional	£ 909
WindowsMaker	£ 599
Zortech C++ Developer's Edition	£ 292

CADD

3D Drafting	£ 200
AutoSketch	£ 143
Design Cad 2D	£ 146
Design CAD 3D	£ 194
DRAFIX CAD for Windows	£ 413
Generic CADD	£ 283

CASE

Case: PM for C++	£ 1810
Case: Windows	£ 903
Compeditor II	£ 227
EasyCASE Plus	£ 255
Oracle CASE Dictionary	£ 1513

COBOL

COBOL spII	£ 418
COBOL/2 Compiler with Toolset	£ 1458
Dialog System	£ 810
Level II COBOL	£ 401
RM/COBOL-85 Development System	£ 760

COMMUNICATIONS

Carbon Copy	£ 110
Crosstalk for Windows 3.0	£ 118
LapLink III	£ 89
Mirror III	£ 80
Reflection 1 for Windows	£ 291
Remote2 (Host & Caller)	£ 105
SmartTerm 240	£ 203
Takeover	£ 159

DATABASE & FILE MANAGEMENT

Advanced Revelation	£ 710
ALPHA/400 competitive upgrade	£ 92
AskSam	£ 213
Btrieve DOS 3.1 Network	£ 386
Btrieve for Windows	£ 386
Clarion Professional Developer	£ 447
DataFlex	£ 396
dbFast/PLUS	£ 168
dbFast/Windows	£ 241
FairCom Toolbox Special Edition	£ 450
Gupta SQL Windows	£ 916
Informix 4GL Rapid Dvlpmt Sys w/ Dbug	£ 698
Informix SQL	£ 524
Paradox SQL Link	£ 337
Paradox v3.5 Competitive Upgrade	£ 136
R&R Clipper and FoxBASE+ Module	£ 42
R&R Relational Report Writer for dBASE	£ 161
R&R Report Writer for Oracle SQL	£ 194
R:BASE Single User	£ 472
Superbase 4 Windows	£ 420
UI Programmer	£ 321
XQL Relational Data Manager	£ 532
Xtrieve Plus	£ 405

DEBUGGERS

ClearView-386/486 Debugger	£ 320
Multiscop Debugger for DOS	£ 114
Multiscop Debugger for Windows	£ 242
Periscope II with Breakout Switch	£ 122
Turbo Debugger & Tools	£ 97

DESKTOP PUBLISHING

Adobe PostScript Cartridge	£ 299
Adobe Type Align /Windows	£ 60
Adobe Type Manager for Windows	£ 61
CatchWord OCR	£ 137
ColorRIX VGA Paint	£ 97
Facelift for Windows	£ 59
Fontware Installation Program	£ 56
Fontware Typeface Packages	£ 77
Go Script Plus	£ 181
Inset + Hijaak	£ 161
OmniPage Pro	£ 602
OmniProof for DOS	£ 99
PageMaker	£ 496
The Complete Page Scanner w/WordScan	£ 583
WordScan 386/386 2 Mb	£ 386
Xerox Ventura Publisher Gold	£ 472

DISK/DOS UTILITY

Back-It	£ 75
Copy II PC	£ 26
Disk Manager	£ 35
Lotus Magellan	£ 98
Norton Commander	£ 95
Norton Utilities	£ 110
OPTune	£ 43
PC Tools Deluxe	£ 102
Q-DOS II	£ 43
SpeedStor Universal	£ 62
WonderPlus	£ 48
XTree Pro	£ 54

EDITORS

Brief	£ 178
EDIT +	£ 214
EDIT	£ 97
Norton Editor	£ 48
Sage Professional Editor	£ 191
VEDIT PLUS	£ 110

FORTRAN

Fortran Toolkit 2	£ 116
GRAFATIC	£ 88
Lahey F77L	£ 482
RM/FORTRAN Development System	£ 347
Spindrift Utility Library (F77L)	£ 80
ULI Mathematical Fcncts Library (F77L)	£ 189

GRAPHICS

Corel Draw	£ 375
Designer for Windows	£ 458
Draw Applause	£ 265
GRASP with ARTTOOLS	£ 192
GSS GKS Kernel System for DOS	£ 418
GSS Graphic Dvlpmt Toolkit for DOS	£ 558
HALO	£ 228
Halo Windows Toolkit	£ 341
Harvard Graphics	£ 302
Hijaak	£ 129
MetaWINDOW	£ 223
PCPaintBrush for Windows	£ 52
Pizazz Plus	£ 85
Windows Draw Plus	£ 161

HARDWARE

8087-10 Math Coprocessor	£ 158
ASCI Internal 2400 Modem	£ 68
ASCI Int 9600bps V.32 w/ MNP 5	£ 431
FastMath 83S87-16 Math coprocessor	£ 244
Hardcard IIXL 105Mb, 9ms Hard Disk	£ 559
Logitech TrackMan Serial Mouse	£ 77
MouseMan Serial Mouse	£ 68

Pacific Page PS Cartridge for HP IIP	£ 350
Paradise 1024x768 VGA w/ 512k	£ 192
Practical 2400 Pocket Modem	£ 106
ProScan PS1000 64 Grayscale Scanner	£ 523
ProScan PS3000 Color Scanner -PC BUS	£ 729
SatisFAXtion Classic	£ 377
ScanMan Plus PC	£ 172
The Complete Communicator	£ 415
The Complete Page Scanner	£ 175
Viva 2400/9600 Send/Receive Fax	£ 194
VIVA 9600 V42 bis External Modem	£ 449
Weitek 3167/25MHz Math Coprocessor	£ 562

INTEGRATED

Framework III	£ 426
Lotus Symphony	£ 488
Lotus Works	£ 90

LAN & LAN READY

LANtastic AE-2 Ethernet Starter Kit	£ 491
Netware 386 20 user	£ 2341
Oracle for Server	£ 1261
Q-DOS II for Network (8-user)	£ 135
Saber File Manager	£ 201
Saber LAN Administration Pack	£ 353
Saber Menu for MS Windows 3.0	£ 250
Saber Menu System Single-User	£ 45
Saber Meter Usage Tracking Software	£ 99
SoftwareBridge LAN	£ 143
Superbase 4 Windows Network	£ 602
Windows eMAIL 1 User MHS Remote	£ 101

MAC

Adobe Illustrator	£ 334
Bernoulli File Exchange	£ 92
Claris CAD	£ 563
FastBack MAC	£ 114
Generic Cadd Level 1 for Mac	£ 86
HyperDesk v4.2	£ 45
LapLink Mac Connectivity Pack	£ 86
LetraStudio	£ 270
MacAstronomy	£ 43
MathCad for MAC	£ 307
Mathematica for Mac	£ 428
MathType	£ 113
Microphone II	£ 204
MinCAD+	£ 465
Norton Utilities for Mac	£ 82
OmniPage (OCR)	£ 428
Oracle Developers Kit Mac	£ 504
Pagemaker	£ 490
PC-Matlab	£ 756
Practica Musica	£ 68
Quark Xpress 3.0	£ 515
Smalltalk/V for Mac	£ 108
StatView II SE+ Graphics	£ 189
Theorist	£ 238
Visual Business No. 5	£ 256

MEMORY MANAGEMENT

386 Max	£ 70
Above Disk	£ 58
QEMM Expanded Memory Manager 50/60	£ 57
QEMM Expanded Memory Manager/386	£ 57
Vcache	£ 45

MODULA-2

EmsStorage with Full Source	£ 48
Logitech Modula-2 Development System	£ 159
Logitech Modula-2 Window Pack	£ 31
ModBase	£ 21

OBJECT ORIENTED PROGRAMMING

Actor for Windows with Language Extns	£ 107
C++ Views	£ 321
Codebase++	£ 223
HCR/C++	£ 731
Smalltalk/V	£ 55
Smalltalk/V 286	£ 108
Smalltalk/V Communications	£ 27
Smalltalk/V EGA/VGA Colour Pack	£ 27
Smalltalk/V Goodies	£ 30
Smalltalk/V-Windows	£ 270
Zinc Interface Library	£ 161
Zortech C++ Compiler	£ 130
Zortech C++ MS-DOS 386	£ 645
Zortech Windows Toolkit	£ 45

OPERATING ENVIRONMENTS

386/ix Operating System Release 3.2	£ 332
DESQview	£ 77
DR DOS Competitive Upgrade	£ 68
DR Multiuser DOS	£ 450
GeoWorks Ensemble	£ 106
Microsoft DOS v5.0 Upgrade	£ 65
Microsoft Windows	£ 93
OS/2 Extended Operating System v1.3	£ 667
OS/2 Standard Operating System v1.3	£ 259
SCO UNIX V/386 Operating System	£ 640

OS/2

Btrieve OS/2	£ 386
Corel Draw for OS/2	£ 321
GSS Graphics Dvlpmt Toolkit for OS/2	£ 558

Halo for OS/2	£ 341
Microsoft Windows Library for OS/2	£ 162
Multiscop Debugger for OS/2	£ 286
Norton Guides for OS/2	£ 94
Oracle OS/2	£ 655
Paradox OS/2	£ 464
PolyAWK for OS/2	£ 129
R:BASE for OS/2	£ 551
Smalltalk/V-PM	£ 270
Vitamin C for OS/2	£ 321
Wide Angle for OS/2	£ 63
Xtrieve Plus for OS/2	£ 399
Zortech C++ Developer's OS/2 Edition	£ 389

OTHER

ACT! (Sales Leads tracking)	£ 235
Adlib Card (Sound)	£ 105
Check-it (Diagnostics)	£ 79
Encore (Music)	£ 360
Globa Link English to French	£ 647
graphics-MENU (GUI)	£ 135
HP NewWave (GUI)	£ 105
HyperPAD (Hypertext)	£ 82
Labels Unlimited (Labels)	£ 36
Menuet (GUI Development)	£ 211
MS Programmer's Library (CD-ROM)	£ 244
Norton Guides	£ 70
PolyAWK	£ 64
PolyLibrarian (Source Control)	£ 64
Polymake	£ 116
Pro-Cite (Bibliography)	£ 256
PVCS Professional (Version Control)	£ 321
RPG II Development System	£ 1123
SCHEMA-PCB (PCB)	£ 737
SoundBlaster (Sound)	£ 148
System Sleuth (Diagnostics)	£ 72
VirusCure Plus (Virus)	£ 65

PASCAL

Asynch Plus	£ 123
B-Tree Filer	£ 98
Developers Library	£ 247
Graf/Drive Plus Personal	£ 80
Object Professional	£ 133
Pascal TOOLS	£ 81
POWER TOOLS PLUS	£ 97
Topaz	£ 65
Turbo Analyst	£ 91
Turbo Pascal for Windows	£ 162
Turbo Pascal Professional	£ 187
Turbo Professional	£ 91

SCIENCE & ENGINEERING

ChiWriter Chemistry Support	£ 45
Derive v2.0	£ 184
InGra	£ 265
Labtech Notebook	£ 806
Matfor - 286	£ 227
Mathematica 386	£ 601
Mathematica for Windows	£ 860
PlotLib	£ 189
SPSS/PC+ Base	£ 152
SSP/PC Scientific Subroutine Library	£ 246
Tech*Graphic*Pad	£ 337

SCREEN DESIGN & MANAGEMENT

C Windows Toolkit	£ 86
Flash Tools	£ 77
Hi Screen XL	£ 113
POWER SCREEN	£ 97
ProPak	£ 415
SayWhat?	£ 33
Vitamin C	£ 256
WINDX GUI Development System	£ 323

TRANSLATORS

BAS C Commercial	£ 243
BAS_PAS Commercial	£ 181
dBx/Dbport with Source	£ 810
FOR C with Runtime Source	£ 632
For Struct	£ 535
Pascal to C Translator	£ 140
Turbo Plus	£ 140

UNIX/XENIX

Architect Series Workstation Developer	£ 1336
Eiffel for UNIX/XENIX	£ 1005
Informix C-ISAM (SCO 4.0)	£ 297
LPI COBOL Compiler	£ 1082
Menuet for UNIX	£ 421
Norton Utilities for System V v3.2	£ 175
Oracle UNIX/Xenix 386 1-user	£ 1486
SCO Open DeskTop	£ 1119
SCO UNIX System V/386 dev Sys	£ 645
VEDIT PLUS for UNIX/XENIX	£ 185
Zortech C++ for SCO Unix	£ 324

WINDOWS DEVELOPMENT

Asymetrix ToolBook.....	£ 282
Becker Tools.....	£ 73
Microsoft C Compiler with Windows SDK.....	£ 469
MS Windows Dev Driver Dev Kit.....	£ 324
Microsoft Windows Productivity Pack.....	£ 38
Object Vision.....	£ 87
WIN++.....	£ 161
WindowsMaker Professional.....	£ 750
WinSleuth.....	£ 72

'backtrack', and undo some task and resource choices and try again using a slightly different strategy. This cycle will continue until the problem is solved. The trick is that, at each cycle, the order of the tasks can change quite significantly. It is in calculating these sets of possibilities that constraint programming is so effective. The decisions made in the previous cycle ripple through all the tasks and resources, and the solution sets are immediately available for the next.

Figure 1 shows the main cycle of this program written as a LISP function.

Most of the burden in getting the tasks in the right order is now placed on the functions which calculate the impact for a solution and the criticality for a task. Getting these functions right requires a lot of care, but the most important point is that they can be fairly rough rules and still be effective. The criticality, for instance, could just be the reciprocal of the number of solutions. It could even do some kind of 'look ahead', much as one might expect to find in a chess program, but this would probably be so much slower that it would be of little help.

Some of the thornier problems are still hidden from this template code. Backtracking, for instance, is quite difficult. If there are no solutions for a task, we could go back to a task where there was more than one solution and pick the second alternative, then the third, and so on. Even better, we could try to work out where the block is, and go straight back to that point. This is known as 'dependency-directed backtracking'. While this method is intended to minimise the amount of backtracking needed, for complex problems some backtracking will probably always be necessary. This approach combines ways of avoiding problems with strategies for curing them once they have occurred. Of course, inevitably it is much better to avoid them if possible.

A statistical heuristic like this has the advantage of being independent of the problem to be solved. In any real scheduler, it would also be worth looking for any more specific heuristics, which can be used to help guide the search for a solution. Any extra ones can usually be added into the calculations of the task criticality and solution impact.

Heuristics like this help achieve a real solution for what would otherwise be intractable problems, but they do not represent the whole story. As with many systems today, much of the effort lies in the human computer interface. In the old days, scheduling systems ran on a mainframe, usually

as overnight batch programs. More recently Graphical User Interfaces have transformed the look and feel of computers. In addition, they have provided a much higher degree of user interaction and thus added many more 'opportunities'. Programs must now

***There are 4.7 x
10¹⁵⁹ different
ways the tasks
can be ordered -
in LISP, this is
known as a
'bignum'!***

be able to respond dynamically to the users requirements and actions - say removing or adding new constraints. This is where a special purpose constraint programming language really comes into its own; new

constraints can be added by the user, and will immediately affect the choices of tasks and solutions. Nothing else needs to be done.

Constraint programming and object-oriented programming are not the same, but they can be combined very neatly. It is easy to extend constraints so they can apply to slots in an object; in some cases this is easier than other kinds of variable. AI purists will even implement a constraint as an object, with methods describing how it constrains the variable values.

The N-Queens Problem

The *n-queens* problem is one of the classical AI problems, and tends to be shown in all sorts of places as an illustration of planning and scheduling. The problem is quite simple - that of arranging *n* queens on an *n* by *n* chess-board such that none of them attack each other. Usually the problem involves an eight by eight board or something bigger.

In terms of scheduling, each queen can be thought of as a task, and each column as a time slot. A very simple solution to the

```
;;; The N-Queens Problem

(defun n-queens (size)
  (catch 'solution
    (n-queens-internal size size ())))

(defun n-queens-internal (size rows-left board)

  ;; When there are no rows left to solve, we have found a solution.
  (when (= 0 rows-left)
    (throw 'solution board))

  ;; For each column, if it does not attack any already on the board,
  ;; the place it and try to place the following rows...
  (dotimes (i size)
    (unless (n-queens-attack-p size i board)
      (n-queens-internal size (1- rows-left) (cons i board)))))

(defun n-queens-attack-p (size column board)

  ;; They attack if another row has the same column.
  (or (member column board)

      ;; Or if any attack on diagonals.
      (let ((offset 1)
            (columns board))
        (loop

          ;; Stop at the end of the board
          (when (null columns)
            (return-from n-queens-attack-p nil))
          (when (= offset (abs (- (first columns) column)))
            (return-from n-queens-attack-p t)))

        ;; Skip to next column, and increase the count of the distance
        ;; between the new column and this one.
        (incf offset)
        (pop columns))))))
```

Figure 2 - The N-Queens Problem

"With or without a mouse, it's the cat's whiskers."

New development environments and technology transition make 80's editors inefficient and obsolete. To be competitive, programmers must have leading edge tools. The tool you'll use the most, the one that wraps everything into a single consistent environment will be the powerful, configurable SPE Professional Editor.

Right out of the box, it'll make you more productive, more adaptable, more imaginative. Its instant installation, elegant mouse support, advanced user interface, and point-and-shoot help get you running immediately.

**MS-DOS, OS/2
and Dual Mode together**
MS-DOS, OS/2 and Dual
Mode versions are all included

in the same box on both 5 $\frac{1}{4}$ " and 3 $\frac{1}{2}$ " disks, with templates for popular languages, with or without a bundled Microsoft® Mouse.

Great value, and your team can move into the 90's with the same editor.

Turnkey emulations

If you prefer the commands and keystrokes of other editors, our turnkey emulations duplicate them precisely. And you still gain the SPE Professional

Editor's advanced features, windowing capabilities and powerful engine.

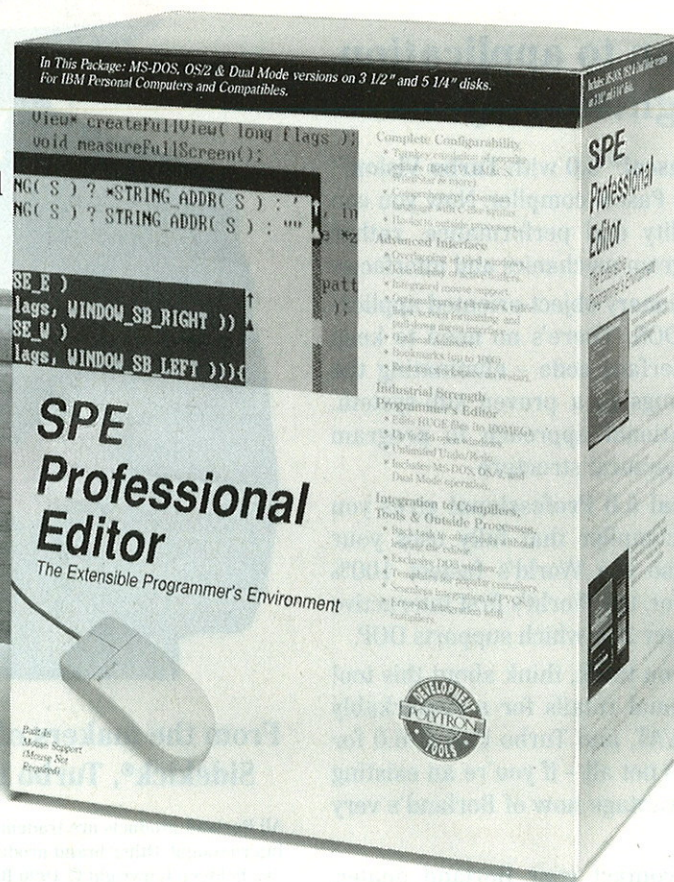
With or without a mouse

Use the editor with or without a mouse — all functions are available without lifting your fingers from the keyboard. But the click-select windows, scroll bars, zoom, shrink-to-icon, block text selects, and other speedy mouse actions make editing extremely efficient.

Don't take our word for it

We're confident that the SPE Professional Editor is the world's best programmer's editor. But don't take our word for it.

Order your copy and decide for yourself.



Works with or without a mouse.
Packaged with or without Microsoft® Mouse.

Version 1.1 now available

APPROVED DEALERS

Absolute Computing Limited, Absolute House
124 Leavesden Road, Watford, Herts. WD2 5EG. Tel: (0923) 242570

Computer Solutions Limited, Canada Road, Byfleet
Surrey KT14 7HQ. Tel: (09323) 52744

Grey Matter Limited, Prigg Meadow, Ashburton
Devon TQ13 7DF. Tel: (0364) 53499

Roundhill Computer Systems Limited, Orchard House
Ogbourne St. George, Marlborough, Wiltshire SN8 1SU. Tel: (0672) 84535

Reflex Technology Limited, 9 Buckingham Place
Bellfield Road, High Wycombe, Bucks HP13 5HW. Tel (0494) 465907

Frontline Distribution Limited, Intec 1, Wade Road
Basingstoke, Hampshire RG24 0NE. Tel: (0256) 463344

System Science, 3-5 Cynthia Street
London N1 9JF. Tel: 071-833 1022

Software Express, Portmill House, Portmill Lane
Hitchin, Herts SG5 1DJ. Tel: (0462) 422525

Software Generation • Portmill House, Portmill Lane, Hitchin, Herts SG5 1DJ

BRAIN BOX

From inspiration to application – the brightest way

BORLAND'S new Turbo Pascal® 6.0 with Turbo Vision™ is the World's Number One Pascal compiler. Now you can concentrate on functionality and performance, rather than spending time on program mechanics and interfaces.

Turbo Vision is a revolutionary object-oriented application framework for MS™-DOS. There's no need to keep writing and debugging interface code – eliminating the risk of introducing new bugs to a proven sub-system. Turbo Vision brings a rational approach to program design with its single architectural structure.

Even better, **Turbo Pascal 6.0 Professional** gives you not only a Turbo-driven compiler that fully uses your extended memory, but also the World's fastest 100% MASM-compatible assembler, the World's first interactive profiler, and **Turbo Debugger 2.0**, which supports OOP.

If all that doesn't make you think, think about this too! Turbo Pascal 6.0 Professional retails for a remarkably competitive £199.95 plus VAT, and Turbo Pascal 6.0 for £99.95 plus VAT. And that's not all – if you're an existing Turbo Pascal user, take advantage **now** of Borland's very special upgrade offers.

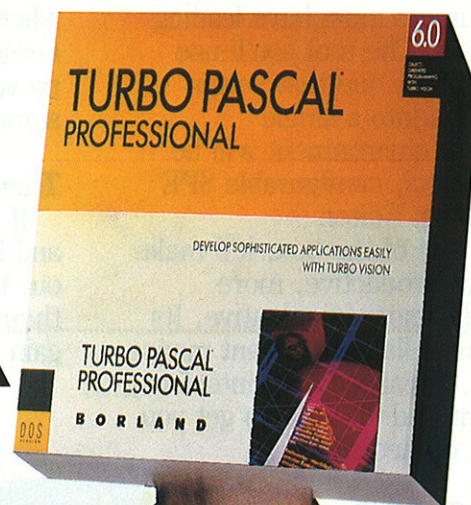
For more information, contact your Borland dealer, ring Borland Customer Services on 0734 320022, or send the Freepost coupon today.

You'll never make a smarter move!

B O R L A N D
S O F T W A R E
K N O W H O W

Borland International (UK) Ltd
8 Pavilions, Ruscombe Business Park
Twyford, Berkshire RG10 9NN Telephone 0734 320022

CIRCLE NO. 972



From the makers of Paradox®, Quattro® Pro, Sidekick®, Turbo Pascal® and Turbo C++®

All Borland products are trademarks or registered trademarks of Borland International. Other brand product names are trademarks of their respective holders. Copyright © 1990 Borland International Inc.

NAME	_____
JOB TITLE	_____
COMPANY	_____
ADDRESS	_____
POSTCODE	_____ TEL _____
I currently use Turbo Pascal <input type="checkbox"/>	
Please return this coupon to: Turbo Pascal 6.0 Information, Borland International (UK) Ltd, FREEPOST RG1571, Twyford, Berkshire RG10 8BR	



n-queens problem is shown in Figure 2. It is a simple blind search, and as a result the performance degrades very badly as the size of the board increases. For eight by eight, it takes about 0.25 seconds, with 12 by 12, it takes about one second, and for 16 by 16 it takes about a minute. Scheduling might easily have 1000 of these to task, so blind search is obviously not good enough.

A second factor is that this solution completely ignores the question of how to resource a task, and this is a significant problem to solve for each task. There is more to placing a task than simply looking to see if it conflicts with any others on the time alone. Each task will have its own search problem, just to choose which resources are best to use, and whether they conflict with the resources needed by tasks already placed.

One important point about the n-queens solution shown in Figure 2 is that it is incremental. Each time round the loop it only checks the changes once. The rest of the board state is not rechecked. This is an important and relatively easy way of saving a lot of time.

Conclusion

We have attempted to show that scheduling problems can be approached in a way which reduces the number of possible alternatives. This makes it possible to cut drasti-

**Deadlock arises if
all the
philosophers go
to the table
simultaneously...**

cally the time taken to explore other solutions. Constraints also offer a powerful method of trimming the search space, and thus increasing the efficiency of the scheduler.

Finally, we hope to have convinced you that LISP is an intrinsically better language for building a constraint based scheduling

system, primarily because (as AI suggested in his earlier article on LISP - *The Third Side*, .EXE December '90) it is so good for implementing 'special purpose programming languages'. This facility allows the provision of higher-level scheduling languages sitting on top of LISP - and these make programming with constraints so much easier.

EXE

Al Roth is a freelance writer, technology consultant, and Carbon-based humanoid biped, with a strong predisposition towards Delicatessens. He can be contacted at al-roth@cix.

Stuart Watt now works for the Open University, having first become interested in computers after a bad fall when young, and has a burning desire to build AI systems in FORTRAN IV. He can be reached at snk_watt@uk.ac.open.acs.vax.

References: The Structure and Interpretation of Computer Programs (by Abelson and Sussman, pub MIT Press) and A Planning/Scheduling Methodology for the Constrained Resource Problem (by Keng and Yun, Proc. 11th IJCAI, Morgan-Kaufman)

"How Did We Ever Manage Without SMS?"

ASM (Europe)

SMS provides a comprehensive set of tools for automating the process of managing different versions of files and configurations of systems. SMS improves *quality, reliability* and *productivity*.

Version 2.1 now available! Even more power and flexibility

Software Management System provides a fully integrated environment; all features can be accessed through a menu-driven Front-end with context-sensitive on-line help.



Version Control

Management of multiple revisions, lines of development and users

- Revision Storage and Retrieval
- Edit Control
- Revision Identification and Reporting
- Minimisation of Storage
- Text and Binary File Support

Configuration Management

Management of configurations of the items which make up a project/product

- Release and Configuration Identification
- Automated System Building
- Variant/Configuration Maintenance
- Automated Dependency Generation

Modification Requests

Formalisation and tracking of bug reporting and upgrade procedures

- Change Management
- User-definable Life-cycle
- Active Notification System
- Modification Analysis and Reporting

Available for: MS-DOS, OS/2, Unix, VMS, OS-9
Single User PC Price £ 490.00 + VAT ; 5 User PC Price £ 980.00 + VAT
Please contact us for further technical and pricing information

INASOFT : the productivity tools for professional software developers
Tresco House, 153 Sweetbrier Lane, Exeter, EX1 3DG, England, Tel: 0392 217670, Fax: 0392 437877

MS-DOS Version 5 - A Quantum Leap or Just Catching Up?

Microsoft launched its latest OS at an extravaganza hosted by TV personality Jonathan Ross. Mark Hamilton, who has been using MS-DOS 5 for two months, can offer a more considered view.

I first heard about Microsoft's DOS 5 last spring (1990) around the time Digital Research launched its DR DOS 5. Fully compatible with Microsoft's 3.31 version, Digital Research had broken new ground by providing users, with 286 or better PCs, with more than 600 KB free after the operating system had loaded.

DR had achieved this by allowing the operating system to relocate the bulk of itself into the 64 KB of memory starting at 1 MB - an area known as the High Memory Area. There was nothing revolution-

ary about this as we shall see, but Digital also built-in the ability to relocate TSRs and Device Drivers into Upper Memory Blocks, situated between the display adapter and the PC's ROM BIOS, without the need to buy third-party memory managers.

On the PC, Microsoft was the first to exploit the use of the High Memory Area with Windows/286 but it was Digital Research which perfected the art of the writing relocatable operating systems - remember CP/M and CP/M-Plus? But I digress.

From a DOS 3.x or 4 user's perspective, there are similarities between the rival DR DOS 5 and MS-DOS 5, but there are also a number of significant differences. On 286 and better PCs with more than 1 MB of memory, both provide more than 610 KB of free memory once the operating system has loaded, with MS-DOS having a slightly smaller conventional memory footprint. Both provide graphical shell programs but Microsoft's includes a task switcher.

The DOS 4.x undocumented TRUENAME command is in DOS 5; it displays the canon-

```

;+=====+
;+ Sample Interrupt 2fh Handler to make a (TSR) program task- +
;+ switcher aware. This code handles both Windows Enhanced and +
;+ Windows Standard / DOS 5 type task-switchers. +
;+=====+
;+ By Mark Hamilton for .EXE Magazine, July 1991 Edition. +
;+=====+
OurInt2fHandler proc far ; Installed via Int 21 (AX=252fh)
    cmp ax, 1605h ; Windows Enhanced Mode Call?
    je OurHandler ; It's for us
    cmp ax, 4b05h ; Windows Standard Mode / Dos 5?
    je OurHandler ; It's for us

; Here could go code to handle other requirements of the Int 2f
; handler.

I2F_Next:
    jmp far dword ptr OldInt2fHandler
; Saved from an Int 21 (AX=352fh) call

; Here if a Task Switcher Identify Instance Data Call. The
; protocol requires that we push the Flags Register on to the
; stack and call the (original) Interrupt 2F handler. In this
; implementation, this is achieved by a simulated call.
OurHandler:
    pushf ; Push Flags on Stack
    push cs ; Simulate a far call.
    push offset SaveContext ; The address we want returned to
    jmp short I2F_Next ; Do the "call"

; Here could go code to handle other requirements of the Int 2f
; handler.

; This is the "Identify Instance Data" handler. Instance data is
; the area of data that needs to be preserved and could be
; changed by one of the various tasks. If CX <> 0, then the
; calling task switcher is either Windows in Standard Mode's or
; DOS 5's task switcher; otherwise, it is Windows Enhanced
; Mode's.
SaveContext:
    jcxz SaveContext_1 ; No call-back address to save
; Offset of Task Switcher Call-In...
    mov word ptr cs:TS_CallIn, dx
    mov word ptr cs:TS_CallIn+2, cx ; ...and segment.

; If called by Windows in enhanced mode, ES:BX will point to a
; previous handler's Start Up Information Structure. In
; Windows Standard Mode and DOS 5, ES:BX is 0:0 (but is
; reserved for future expansion). Save it in either case.
; Return with ES:BX pointing to our Start Up Information
; Structure (SIS).
SaveContext1:
    mov word ptr cs:SIS_Next_Dev_Ptr, bx
    mov word ptr cs:SIS_Next_Dev_Ptr+2, es
    push cs
    pop es
    mov bx, offset cs:SIS_Version
    iret

; This is the Start Up Information Structure. Certain fields
; (marked "**") must be completed by the program's
; initialisation procedure.
SIS_Version db 3,0 ; Major, minor version
SIS_Next_Dev_Ptr dd ? ; Ptr to previous handler's SIS
SIS_Virt_Dev_File_Ptr dd 0 ; Ptr to ASCIIZ name of
; virtual device name, or zero
; (ignored here)
SIS_Reference_Data dd 0 ; Ptr to virtual device,
; non-zero only if
; SIS_Virt_Dev_File_Ptr is
; non-zero (ignored here)
SIS_Instance_Data_Ptr dd IIS ; Pointer to Instance Item
; Structure (*)

; There now follows the Instance Item Data Structure. This
; comprises a far pointer to the beginning of the data block
; followed by that data block's length. This can be repeated as
; many times as necessary and is terminated by a double-word
; zero.
IIS dd BeginData ; Ptr to Data Block 1 (*)
dw Block1Len ; Length of
; Data Block 1 (*)
dd 0 ; End of IIS

; The Standard Mode switcher's Call In Address should be saved
; here.
TS_CallIn dd 0 ; Non-Zero if Standard
; (DOS 5) mode.

OurInt2fHandler endp

```

Figure 1 - DOS 5 Sample Interrupt 2fh Handler

5 out of 5 hackers prefer other software protection methods to Hardlock E-Y-E®



What hackers dislike...

Hardlock E-Y-E was designed using cryptographic principles. It took the experience and know-how of Germany's No. 1 in software protection and the leading edge technology of a US semiconductor company to create the ultimate software protection tool. Hardlock E-Y-E is based on a custom chip featuring secure algorithmic response rather than simple bit swapping or counting schemes.

What software developers like...

Hardlock E-Y-E combines all the features software developers require in a single product: algorithmic response to provide security and an optional non-volatile memory to allow custom configurations. FAST Electronic has made implementation of Hardlock E-Y-E in your software easy. Use HL-Crypt to protect .EXE or .COM files, or incorporate high level language interface routines in your software. The algorithm parameters and the contents of the memory can be programmed in seconds using our Crypto-Programmer card. This unique card guarantees that no one else can burn your original codes. Simply plug the card into any PC slot and start up your own Hardlock E-Y-E workshop.

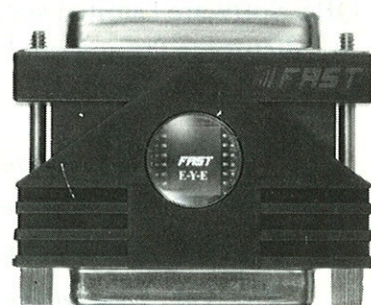
What your customers will like...

Hardlock E-Y-E allows unlimited backup copies. The device is shipped with the software for the user simply to plug into the parallel interface and forget.

Daisy chainability, outstanding reliability (no battery is needed), and the most compact High-Tech design ensure that your customer will accept Hardlock E-Y-E.

What your accountant will like...

Hardlock E-Y-E needs no factory coding. This ensures optimum delivery schedules and stock flexibility. Revenues will go up as software piracy and multiple usage are prevented. Despite its wealth of features, Hardlock E-Y-E's prices remain competitive.



Hardlock E-Y-E
programmable, algorithmic response
and memory option - all in one.

...As more and more software developers, customers and accountants appreciate the Hardlock E-Y-E device, hackers like it less and less.



Order your demo unit today. Contact Magnifeye,

235-239 Walmer Road, Walmer Studio #6, W11 4EY, Telephone 071 221 8024, Fax 071 792 3449.

cal form of a file, taking networked and SUBST'd drives into account. It is functionally equivalent to Interrupt 21 function 60h where DS:SI points to the ASCII file name or path and ES:DI points to a 128 byte buffer for the canonical form. Incidentally, the command 'TRUENAME CON:' displays '/DEV/CON'.

In common with DR DOS 5 and some third-party memory managers, Microsoft has provided the ability for programmers to allocate memory blocks outside the 640 KB limit, specifically in the Upper Memory Area. DOS function 58h (Get/Set Memory Allocation Strategy) now uses the BL register: if bit 7 is set, then the Get/Set strategy applies to Upper Memory Blocks. At the command line level, programs (specifically TSRs) can be loaded and executed in the Upper Memory Area using the LOADHIGH (or LH) command.

The Task Switcher

The task swapper is derived from the Windows 3 Standard Mode task swapper and is keystroke-compatible with it. Thus the [Alt][Tab] key combination cycles through applications, [Alt][Esc] switches to the next program and [Ctrl][Esc] switches back to the DOS Shell pro-

gram. In common with the Windows Standard Mode switcher, it does not support pre-emptive multi-tasking, unlike the switcher in Windows Enhanced Mode.

DOS 5's task switcher presents programmers with a new set of considerations. Suppose you want to write a TSR which monitors the state of the PC and pops up if certain conditions are met - such as a viral attack. Not only do you need to make that TSR aware of pre-emptive multi-tasking environments, such as Windows 3 in Enhanced Mode, but also non-multi-tasking switchers as in DOS 5.

For Windows Enhanced Mode, your TSR would monitor Interrupt 2fh for a call where AX is 1605h. In the case of DOS 5 (and Windows Standard Mode), the task-switcher issues Interrupt 2fh with 4b05h in AX. But, unlike Enhanced Mode, the Standard Mode switchers provide a call-in address which are mainly used by other task switchers. For example, if Windows 3 was invoked in Standard Mode by DOS 5's Shell Program, the former's task switcher would, using the DOS 5 switcher's call-in address, request that the DOS 5 switcher suspend itself while Windows is running.

The code fragment in Figure 1 provides an example of the minimum level of support that a global client program, such as a TSR, has to provide. Things get a lot more complex if your TSR or stand-alone program needs to monitor the task-switcher's activity. Your program could, for example, be in the middle of a critical section which it would be unwise to suspend. In such instances, you can prevent the switcher from suspending your program. But the task switcher's API is too complex to be dealt with in an introductory article of this nature.

Get to know, and use, the Task Switcher API, because I have a feeling that this technology is going to be enhanced as DOS's version number climbs. And Bill Gates has committed Microsoft to enhance DOS further.

EXE

Mark Hamilton is a freelance technical journalist specialising in languages and all flavours of DOS. He has a special interest in computer viruses and is a founder member of the National Computer Virus Strategy Group, setup earlier this year by Scotland Yard's Computer Crime Unit.

CLIPPER ADD-ON SOFTWARE

QBS Software Limited specialises in providing the best add-on libraries and utilities for Clipper. We distribute throughout Europe providing local support centres in France, Germany, Italy and elsewhere through our agents.

Flexfile	Variable length fields	SilverPaint	Graphics Library
Fast Text Search	Advanced Text Search Technology	SubNtx()	Filtering Utility
Dr Switch	Create RAM resident applications	SpellCode	Spell Checker
FUNcky	General Function Library	The Engine	Linkable Spell Checker
Blinker	Dynamic Overlay Linker	Biton	Oracle Library
Netlib	Networking Library	Scripton	Postscript Library
SilverComm	Communications Library	Overlay()	Memory Roll Out Utility
GFORCE Fast graphical interface for Clipper			

All trademarks recognised.

90 days technical support by phone or fax provided on all systems

For further information, free demo software, prices and how to order please contact:

QBS Software Limited, 10 Barley Mow Passage, London W4 4PH
Tel: 081-994 4842 Fax: 081-994 3441
BBS: 081-747 1979

When your system has a weak point . . .



. . . Quintus Prolog is the strength you're looking for.

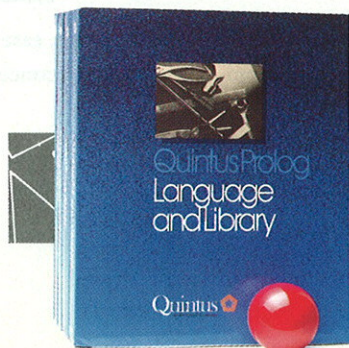
Today's practical systems programmers require robust, powerful and productive software tools and more than just one. No one language offers a complete solution in itself. For example, C is a general purpose programming language offering great portability, whilst Prolog gives enormous programmer productivity for searching databases, manipulating strings, reasoning and matching patterns. Each language is powerful when used individually. When used together their potential is unlimited.

Using each language for what it does best is a good solution to the development efficiency problem. With **Quintus Prolog Release Three** complete integration can be easily achieved using C, C++ and other languages.

You can add power and sophistication to applications without adding reams of code. We can prove it.

Call us, on (0923) 211657.

So avoid weak spots, use the right tools for the job – Quintus and LPA Prolog Products available from A I International Ltd.



ARTIFICIAL INTELLIGENCE INTERNATIONAL LIMITED
Gresham House, 53 Clarendon Road, Watford, Herts. WD1 1LA
Telephone: 0923 211657 Fax: 0923 247836

Now available for UNIX

Zortech's commitment to multi-platform support for C++ now spans MS-DOS, OS/2, DOS 386, UNIX 386 and Macintosh.

John Cant has been looking at Zortech's C++ for SCO UNIX System V/386.

Zortech's C++ for UNIX arrived as three perfect-bound manuals and a single disk. The manuals follow the familiar Zortech style - except being pink - and comprise a chunky 582-page *Function Reference*, a *Compiler Reference* and an *Installation Guide*.

To get it over with, here is the documented spec of the system requirements: an IBM PS/2 or AT compatible computer with a 80386 or i486 processor, 4 MB of RAM, 5 MB of free hard disk space, a 1.2MB or 1.44 MB high density floppy disk drive and the SCO UNIX System V/386 Development System - preferably at least release 3.2.2. The compilers generate 32-bit code to V2.0 of AT&T's C++ standard, including multiple inheritance, type-safe linkage and some V2.1 features, and are designed to maintain a high degree of compatibility with Zortech products available for other operating systems.

Installation

Installation is a simple matter of loading a file from the distribution disk using `install` or `tar`, and uncompressing it. Next, each user's environment files have to be edited to set up correctly variables such as `path`, `include` and `lib`. If the supplied distribution disk does not match your default drive, or you are not using the Bourne shell, then the installation guide is not much help, and you will need to delve into your UNIX documentation.

The system requirements specify that you need the UNIX Development System; this would be unfortunate if true, because the Zortech system should provide everything needed to develop C++ applications. As far as I can determine, however, all that would be seriously missing if you only have the UNIX OS would be `ma sm` the assembler. The UNIX linker `ld`, which links together the Zortech object modules, can be extracted

from the OS link kit by using `custom`.

To complete the installation procedure, I needed two more files which I found on Zortech's Bulletin Board (081 855 3286) -

The limitations of zwb are minor concerns, when the only alternative is the dreaded vi

updates to `ztc` (the compilation control program) and `make`. I downloaded the files under DOS and then `doscped` them into the `/zortech/bin` directory.

I should explain that the system I am running is SCO XENIX 386 Version 2.3.2 on a 386-SX machine with 5 MB of RAM and a 32 MB XENIX partition with both the XENIX OS and the XENIX Development System installed. How much difference it makes using XENIX rather than UNIX is not clear - see my comments below.

Zortech's Work Bench

Perhaps the most immediately appealing aspect of Zortech C++ for UNIX is the provision of a decent editor - the *Zortech Work Bench* (`zwb`). `zwb` allows you to edit, compile and debug C and C++ code without having to leave the editor. Under DOS, I have to admit to not liking `zwb`'s limited keyboard configurability and absence of an

undo facility, and so under DOS have stuck with Zortech's previous editor, `ZED`, configured to look as much like `PC-Write` as possible. Under XENIX, however, these are minor concerns when the only alternative is the dreaded `vi`.

`zwb` seems to be quite happy running on any of the 12 multiscreen terminals (ie virtual terminals) that are available on the console, and one can switch between them using the `Ctrl-Alt-Function` keys. The documentation states that `zwb` will eventually also run on ANSI terminals. I tried running up `zwb` from a number of terminal emulation programs on a separate PC, but each came up with the message '`zwb` does not support terminals smaller than 80 x 25'.

A serious (but presumably temporary) problem is that `zwb` only recognises US keyboards, so that it is vital to keep your brain in full gear while pounding the keys. There are also core dumps if you try to '`shell out`' to the OS.

Hello cliché

Having completed installation, I immediately got to grips with the *de rigueur* 'hello world' program. I created `hello.cpp` and a `makefile`, and then invoked `make` from within `zwb`. No problems, except that XENIX's `make` couldn't make head nor tail of a `.cpp.o:` directive. Once I'd rearranged my path so that `/zortech/bin` was positioned before `/bin`, Zortech's `make` took over, and the time-honoured message appeared on the screen.

```
sh
all_true sleep 5
echo $?
^D
```

Figure 1 - C shell test script

My next thought was to try out some UNIX-specific gobbledegook. An interesting problem appeared in the March edition of the *C Users Journal* - how to get **any** application that is terminated by pressing the delete key to return an exit code of zero.

The author had suggested the following solution: create a process `all_true` which ignores the delete key and spawns a child which then runs the application with the default delete key handling. If the application and child are terminated with 'delete', then the child returns to the parent which exits with status zero.

The program (Figure 2) compiled and ran correctly (if mysteriously), giving the exit code 2000. To determine the exit code, I ran the script given in Figure 1 under the C shell, and pressed the delete key after a couple of seconds. Why the program returns 2000 is anybody's guess, but it's what the CUJ found, so at least Zortech is functioning as it should!

Testing C++ code

This far, I had tested the compiler compiling C code and handling UNIX system calls correctly. My final test was to copy Zortech's database library source files from DOS and to try to compile them under UNIX.

```
#include <stdio.h>
#include <stdlib.h>
#include <process.h>
#include <signal.h>

// forwards
void RunApp( int, char**);

void main( int argc, char** argv)
{
    RunApp( argc, argv);
    exit( 0);
}

void RunApp( int argc, char** argv)
{
    int pid;

    if( (pid = fork()) == -1)
    {
        perror( "all_true");
        exit( 1);
    }
    if( pid > 0)
    {
        /* ignore interrupt key */
        signal( SIGINT, SIG_IGN);
        while( wait( (int*)0) == pid){}
        return;
    }
    /* default interrupt key */
    signal( SIGINT, SIG_DFL);
    argv++;
    execvp( *argv, argv);
    perror( "all_true");
}
```

Figure 2 - Exit on delete key program

```
$4 ~ /-.-/{
    printf( "%s.%s\n", $1,$2);
    system( sprintf( "doscp a: %s.%s .", $1,$2));
}
```

Figure 3 - Awk script for multi-file doscp

I did this by copying the DOS files onto a floppy (although I could have accessed them directly had their DOS partition been bootable), and by using the UNIX `doscp`

***Contrary to the
manual, the
bioskey functions
ARE available -
they do love to
tease at Zortech***

utility. `doscp` is a pain because, with a lot of files to transfer, `doscp a: *` does not work - the command does not accept wild cards. Although UNIX gurus will no doubt

disparage, Figure 3 contains an awk script solution that I find very useful; you just pipe the output of `dosdir a: *` into it. Then, as `dosdir` produces capitalised file names, I wrote a quick `tolower` program to mv these files to lower case names.

Getting the database code to work was the real test of the compiler - handling serious C++ code, and a real test of my tenacity (and the supply of Lymeswold) at porting. After several days of fury at the linker later, I discovered that removing the memory model flag '-ml' from the `ztc` command line solved a multitude of problems.

Zortech UNIX C++ uses a single memory model with a 32-bit address space; in UNIX terminology this is deemed a 'small' model, although to anyone from the DOS world a maximum code and data size of 4 GB seems rather large. It is also worth noting that while `far` pointers are supported by Zortech, apparently the UNIX linker does not handle `far` addresses correctly, and they should not be used. I made some attempts to cause problems deliberately with

```
char far string[] =
    "far data";
```

but was unable to generate a failure.

With suitable hacking at the 'makefile', I was able to get a number of the various programs designed to exercise the database functionality to compile, link and run. The file `irem.cpp`, which under DOS requires the use of the 'DOS extended' version of the C++ compiler, caused a 'shell memory error' during global optimisation, although it compiled fine without optimisation.

The programs that exercise the ISAM file system, however, produced segmentation violations when attempting to add a record. While trying to track this down, I found that if I took up 'too much' stack space with automatics, eg:

```
void main()
{
    short buffer[2*1024];
}
```

then this, too, would produce a segmentation violation. I reported this problem to

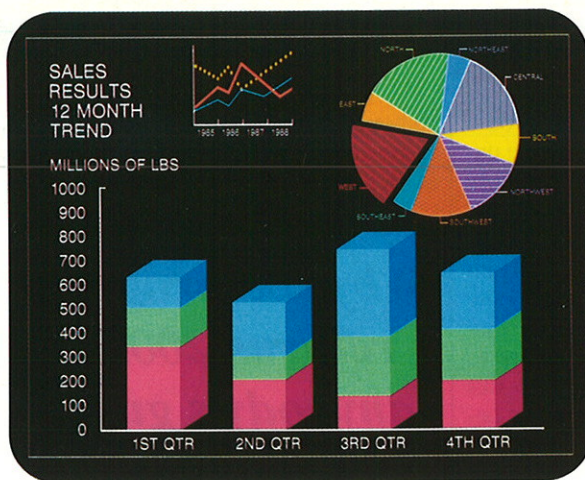
dGE

VERSION 4

GRAPHICS FOR dBASE

**BITS PER
SECOND**

BITS PER SECOND LIMITED
14 REGENT HILL BRIGHTON BN1 3ED
TELEPHONE: 0273 727119
FAX: 0273 731925



dGE extends the database language of your choice to give full colour graphs and charts from **within** your programs.

dGE version 3 was voted the "best graphics tool for the database manager" for the second year running by readers of the influential U.S. magazine DataBased Advisor.

Now Version 4 is here with even more features:

- SuperText with variable sizes and face styles
- SuperIcons loaded from file
- SuperText and SuperIcon editor that can import PCX images
- Built-in Vector Text functions
- Gantt charts
- XOR capability added to many functions
- New interfaces to Clipper 5, FORCE and Vulcan
- Mouse "hot" regions
- A Norton Guide to functions

dGX, the Graphics Design Centre, has been completely revised, adding:

- Sophisticated database query and summary
- Dynamic scaling
- Import of PCX images
- Statistics functions
- Gantt charts, hi-lo and 3-D bars

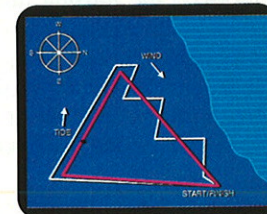
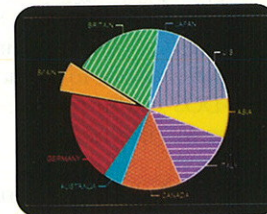
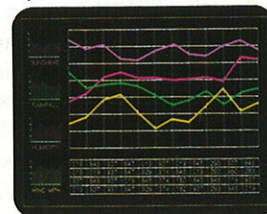
Version 4 comes complete with all the interfaces to dBASE and its dialects, the 'C' language, R:BASE compiler and Vulcan.

dGE supports CGA, EGA, VGA and Hercules modes and will output images to matrix printers, lasers and plotter.

Call for information.

Price £245 + VAT direct from Bits Per Second Ltd or from your authorised dealer.

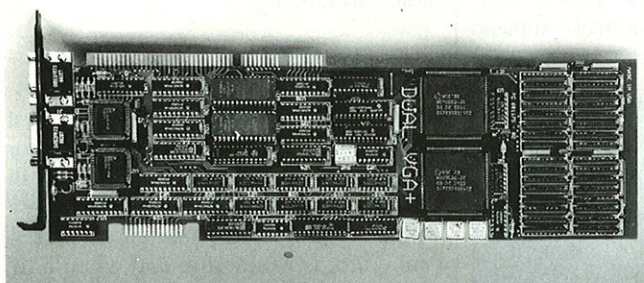
dGE IS A TRADEMARK OF BITS PER SECOND LTD. OTHER BRAND AND PRODUCT NAMES ARE TRADEMARKS OF THEIR RESPECTIVE HOLDERS.



CIRCLE NO. 978

**MULTIPLE INDEPENDENT MONITORS
CAN NOW BE DRIVEN SIMULTANEOUSLY
FROM A SINGLE PC OR PS/2**

THE DUAL VGA PLUS



- ★ UP TO 800x600 RESOLUTION
- ★ TWO INDEPENDENT SIMULTANEOUS DISPLAYS WITH OVERLAY CAPABILITY
- ★ 100% COMPATIBLE WITH IBM VGA/EGA/CGA/MDA VIDEO STANDARDS
- ★ 4 BOARDS IN ONE SYSTEM GIVE 8 SIMULTANEOUS DISPLAYS

**NEW DRIVER! ALLOWS WINDOWS 3.0 TO
BE EXPANDED ACROSS FOUR MONITORS**

CEBRA COMMUNICATIONS LIMITED

26 LORNE PARK ROAD, BOURNMOUTH
TEL 0202 299048 FAX 0202 299192

CIRCLE NO. 979

EMBEDDED

Use Turbo Debugger LOCATE - £304

Locator fully supports Turbo C, Turbo C++ and Microsoft C. Complete with startup code and examples. Emulator and EPROM programmer support.

TDREM - £194

Target based monitor links to Turbo Debugger in remote mode. Supports 80x86, V-Series. Uses only 5k ROM, 2k RAM. Serial drivers supplied and full custom information.

Carriage Free, excluding VAT.

THE AFFORDABLE SOLUTION



All Trademarks
acknowledged

Tel: 0761 52116 Fax: 0761 53226

**GREAT WESTERN
INSTRUMENTS LTD**
32 Paulmont Rise
Temple Cloud
BRISTOL BS18 5DZ

CIRCLE NO. 980

SYSTEMS DESIGN

Zortech, who tried out the problem on a UNIX (not XENIX) machine and were unable to duplicate the error. It therefore looks as though this is a XENIX-specific problem. A Zortech spokesman admitted that the XENIX support of this compiler was something of an afterthought - the company had originally intended the compiler to be UNIX-only, but had yielded to demand. Zortech was looking into this problem but was unable to resolve it before .EXE went to press.

I had hoped to get multi-user versions to compile so as to see my split personalities on different virtual terminals fighting over records, but due to the lack of an `intdos` MS-DOS call to provide file locking, this hope was frustrated.

Had I had more time, perhaps I could have pinned down some of these difficulties and replaced the missing functions with suitable UNIX versions. No doubt whatever problems exist that are not due to my own inadequacies will be rectified shortly, and free updates sent out in accordance with Zortech's excellent support policy.

As Zortech's debugger is not yet available, I was using the UNIX debugger `adb` to have a look under the lid. It is worth noting that to use `adb` effectively, the default `-s` switch has to be removed from the linker command line to enable the generation of a symbol table.

What you get

At present, Zortech's C++ for UNIX is limited to the compiler, the editor `zwb` and the library source. No debugger nor librarian are as yet available, nor UNIX versions of the *Tools* and *Database* libraries.

Although the *Function Reference* manual is huge, documenting over 400 functions, this is because it is clearly an attempt to cover

Why the program returns 2000 is anybody's guess, but at least Zortech is functioning as it should!

all platforms. Figure 4 shows which Zortech Packages are currently included with C++ for UNIX. A number of the standard functions are documented but not supported, eg:

'BIOS - These functions are not available under UNIX, but they are documented here for the convenience of users porting to and from other operating systems.'

However, contrary to this statement the `bioskey` functions are available (they do love to tease at Zortech!), with the proviso that the function `_bios_keybrd_open` must be called beforehand and `_bios_keybrd_close` afterwards. `bioskey` returns the codes that you would get under DOS. One has to be careful using functions like `bioskey` and `getch`, however, because they disable normal command line processing such as `doit < whattodo`.

The Display Package seemed to work (with a few wrinkles) on my VGA display, and also on a VT-100 terminal. Zortech's readme file states that it has not yet been confirmed to work on monochrome or Hercules displays, nor under UNIX Version 3.2. Apparently there is a fault in early versions of SCO UNIX which means that flash graphics routines can only be run as `root` unless you upgrade to SCO UNIX Release 3.2.2

The libraries also contain the following UNIX-specific calls: `alarm`, `chown`, `fork`, `getgid`, `getuid`, `getpid`, `ioctl`, `kill`, `mknod`, `mount`, `pipe`, `ptrace`, `setgid`, `setuid`, `sync`, `umount` and `wait`.

Floating Point

According to the documentation, a variety of options are available for handling floating point maths. If you have a coprocessor or a 486, then the compiler will generate code specific to these. Otherwise software emulation is provided.

For the hell of it, I compiled some `sqrt` and `log` calculations as if I had a 486 or an 80387 - which I don't - and encountered no problems. I found that the Zortech global variable `_8087` was set to a 'co-processor is present' value, so I must remember to open my box to see if one has miraculously appeared.

It is worth noting that if you are doing no floating point maths at all, then the `-mi` switch excludes the floating point libraries, and reduces the size of the executable substantially.

Conclusion

Zortech has produced a creditable first release of a C++ compiler and editor for UNIX. There are problems, particularly in support for XENIX, but no doubt these will be quickly sorted out. Having a decent editor is wonderful, and when fully functional, including Zortech's debugging system, this will be a formidable product.

EXE

John Cant has been an independent software consultant for the last eight years, specialising in support for scientific research programmes and communications in C, C++ and assembly languages. John is an associate of PHD Computer Consultants (033 484 417).

Zortech's UNIX C++ V2.1 compiler costs £299.95 for the Standard Edition, or £399.95 with source. Zortech is on 081 316-7777

Currently supported	
Display Package	fast screen character I/O
Flash Graphics	fast graphics routines for many types of display
File Toolkit	file manipulation
Filespec Toolkit	manipulation of file names
Handle Package	access to virtual memory - defaulted to normal memory allocation under UNIX
List Toolkit	singly linked lists
Mem Package	manipulation of memory buffers, with debug support
Page Package	heap manipulation
String Package	
Time Package	
Not YET supported	
Mouse Package	
Sound Package	
Not supported	
BIOS	
DOS	
FAR Package	memory allocation from the far heap
Interrupt Package	MS-DOS interrupt handlers

Figure 4 - Library Packages

**Windows 3.0 DLL
Now Included!
ROYALTY FREE**

Objective dBASE programmers try C++

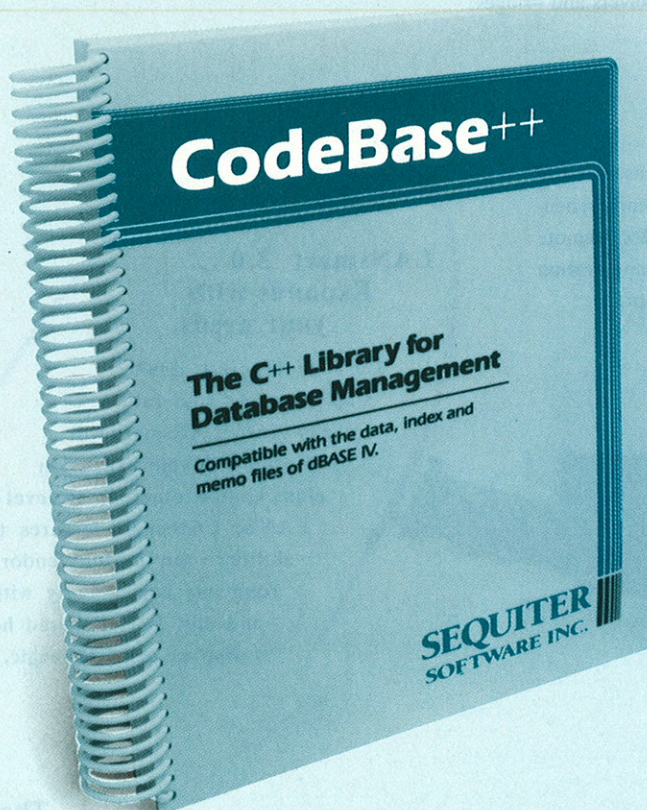
Use CodeBase++ with C++ to build multi-user, dBASE compatible programs. Take advantage of object orientated programming to quickly create maintainable, readable software.

dBASE Compatible

Use CodeBase++ classes to access and change the data, index and memo files of dBASE IV and III. Use other classes to manipulate strings, sort at high speed, perform date arithmetic, evaluate dBASE expressions, manipulate linked lists and manage memory.

Portable

Port your programs between DOS, Microsoft Windows, and OS/2. Keep all the profits as you distribute your programs royalty free.



Fast

Watch as your C++ programs execute much faster than corresponding dBASE, Clipper or FoxPro programs. Be surprised at the small executables and low memory requirements. Examine the C++ source code, which is completely included, to learn good object orientated programming techniques.

Order Today

Order today at £195 by contacting your nearest software dealer. Discover why Sequiter Software Inc. and participating dealers offer a 60-day money back guarantee.

**SEQUITER
SOFTWARE INC.**



**Call 1-403-448-0313
Fax 1-403-448-0315
Hours: 14:30 - 24:00 GMT**

P.O. Box 5659, Station L, Edmonton, Alberta, Canada T6C 4G1

CIRCLE NO. 981



3.0

Network Operating System

LANsmart 3.0 ... The new standard

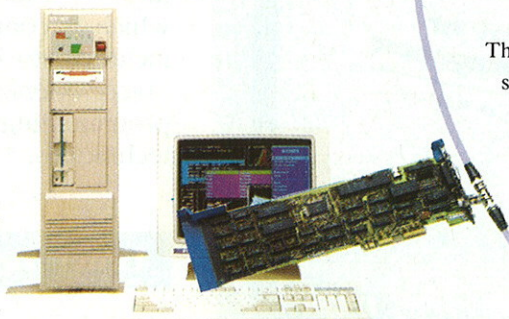
By developing LANsmart, D-Link has redefined the standard for network simplicity, speed and ease of use for entry level users.

LANsmart 3.0 ... Pre-configured LAN packs. Install and expand your LAN in less time, and with less hassle.

For even easier and faster installation, D-Link has introduced complete LANsmart starter kits for 2, 5, 8, 12, or 15 users. Every kit contains all you need to install your network, including LAN cards, cables, etc. Adding new users to the network can't be simpler using D-Link's extensive range of networking hardware for Ethernet, 10BaseT and ARCnet including LAN cards and Pocket LAN adaptors for Laptops and other network accessories such as repeaters, hubs, transceivers and bridges.

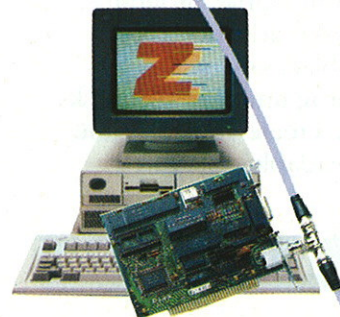
LANsmart 3.0 ... The complete LAN solution.

For more than three years, thousands of users world wide have already selected LANsmart. Its comprehensive range of add-on software options includes: Remote Boot, Electronic Mail, Asynchronous Communication Servers, Remote Access and Screen Monitor.



LANsmart 3.0 ... Expands with your needs.

The benefits just don't stop there. As your future needs grow, LANsmart will still cope with your expansion plans. Unlike other entry level LAN's, LANsmart features the ability to run in Multivendor environments concurrently with Novell and any TCP/IP based host. That is what we call LANmagic...



LANsmart 3.0
The standard others have to follow

D-Link®

Commitment to Connectivity

D-Link (U.K.) Ltd.
Denmark House
Staples Corner
The Broadway
London NW9 7BW
Tel : 081-203 9900
Fax: 081-203 6915

D-Link Systems, Inc., 5 Musick, Irvine, CA 92718
Specifications subject to change without prior notice. All names & trademarks belong to their proprietors.

CIRCLE NO. 982

MASM 6 - Assembly language or 3GL?

*Microsoft's new MASM 6 is claimed to offer performance enhancements over its predecessor.
But that's not all, as Don Binns found out...*

Microsoft's MASM V5.1, long considered the industry standard for macro assemblers, has been in poor health for some time. There always was a somewhat dubious correspondence between the MASM 5.1 documentation and actuality, and recently its facilities have begun to appear unnecessarily limited; but it is reliable, and at one time there was little choice for the assembly programmer looking for a 'safe' macro assembler.

However, for MASM, *de facto* has led to its downfall. Third party assemblers have started to do what MASM does 'with bells on'; frequently better, invariably faster. Borland's TASM provides one of the closest MASM emulations of a third party product, and boasts a clean and attractive 'ideal mode' for users not forced to write MASM compatible code. With version 2.5, thanks to the new two-pass algorithm, TASM outstrips MASM 5.1 in terms of speed, notational richness and optimisation. SLR's Optasm has remarkably humble origins for a product that has gained an increasingly loyal and sizeable following, due to lightning fast compile times and an excellent set of instructions and directives, including a set of object oriented extensions built around its basic but flexible macro language.

Microsoft has responded to this state of affairs with a product intended not only to remedy its sales situation, which was never too dire, but also to overturn received wisdom as regards the rightful domain of a macro assembler. MASM 6.0 represents a significant leap forward in assembler technology.

N-pass assembler

Optasm has lost its position as the fastest DOS and OS/2 macro assembler, chiefly

because of Microsoft's use of an *n-pass* algorithm. In this technique, the first pass is similar to that of traditional assemblers, but the additional information, that normally requires a second or a third read of the

To dismiss MASM 6.0 as a better MASM 5.1 would be to do Microsoft a great injustice

disk-based source, is retained in memory. Several difficulties encountered in traditional multi-pass assembly are avoided in this way. Phase errors are eliminated, as are assembler generated NOPS. This is because the instruction size is not fixed at the first pass. The internal representation of the code is permitted to alter in size, until each instruction reaches its optimum length. Because it references the disk only once, MASM 6.0 assembles two to three times faster than MASM 5.1 - despite the increased optimisation.

One area of improved optimisation is the jump length 'tweaking' that takes place. The assembler determines the smallest encoding possible for a direct unconditional jump. These 'smart' jump optimisations make it unnecessary to specify the distance, which is automatically determined as MASM makes its multiple passes over the in-memory image. Jumps may be lengthened or shortened, as

circumstances dictate. FAR calls made to a destination in the current code segment are automatically optimised by pushing the CS register and generating a NEAR call. Be warned, though, if the programmer does specify a distance and it is too short, the assembler will generate an error. There may be occasions when jump optimisations are not wanted. In these cases, and as a comfort to the wary, `OPTION NOLJMP / LJMP` can be used to switch off these optimisations.

Expanded state control

Several new directives enable or disable various aspects of the assembler control, such as the new 80486 coprocessor option instructions and the use of compatibility options. One such is the new `OPTION` directive, analogous to the `#pragma` directive in C, which allows the user selectively to define the assembler's behaviour, including the enabling of MASM 5.1 compatibility.

The `OPTION` directive is also used to specify the language calling/naming convention that is used. This may be C, BASIC, PASCAL, FORTRAN, SYSCALL or STDCALL. The last two are of special interest: SYSCALL is a language type new to the forthcoming OS/2 V2.0. It is identical to the C calling routine, except no preceding underscore is added to function/procedure names. STDCALL causes the responsibility of stack clean-up to be placed on the calling routine if the `VARARG` keyword is specified. Normally it is the called routine that must clean up the stack.

The new scoping operators give the code a distinctly C++ style, with the same double colon '::' operator. The `SCOPED` option causes any label defined with the tradi-

@Startup:	shl bx, 1
mov dx, DGROUP	cli ;not necessary in .286 or higher
mov ds, dx	mov ss, dx
mov bx, ss	add sp, bx
sub bx, dx	sti ;not necessary in .286 or higher
shl bx, 1 ; if .286 or higher, this is	
shl bx, 1 ;shortened to shl bx, 4	
shl bx, 1	END @Startup

Figure 1a - .STARTUP code in DOS mode with NEARSTACK

tional label: syntax to be local to the procedure in which it is defined. Labels defined with the label:: syntax, or the LABEL directive, are global to the file. The NOSCOPE option is enabled with the M510 option (MASM 5.1 compatibility mode), since previous versions of the assembler did not have scoped labels. While by no means object-oriented, these extensions are part of the new feature set which makes an object-oriented style possible without tortuous macro definitions.

Stack frames

Of especial interest are the new argument keywords, which provide total user control over the generation of procedure prologues and epilogues. EPILOGUE: macroname instructs the assembler to call the macroname to generate a user defined epilogue (instead of the standard epilogue) when a RET is encountered. PROLOGUE: macroname behaves in the same way for prologues. Under the right circumstances, these allow ultra-compact code to be produced, with minimal epilogue and prologue code generated. User-defined stack frames are a powerful feature, with uses limited only by your imagination. This is an example of Microsoft making generally available a feature that has been used in most significant Microsoft products of the last few years.

New directives

Even the most fleeting examination of the new assembler reveals extensive use of automatic code generation from special directives. Apart from the obvious gain in code simplification and ease of maintenance, the heavy use of high-level directives also makes possible a unified assembler interface under a multiplicity of environ-

ments, operating systems and even CPUs. Dependence on these new directives should lessen the impact on low-level programmers of the NT kernel, and the resultant release from the thrall of the Intel segmented processor.

Optasm has lost its position as the fastest DOS and OS/2 macro assembler

A minimal MASM 6.0 program looks like this:

```
.CODE
.STARTUP;generates code
;no code body
.EXIT;generates code
END
```

This example does nothing useful at all, but code is generated by the two directives .STARTUP and .EXIT. The instructions that are generated depends on the target processor (Intel or NEC) and the operating system specified after .MODEL directive. A DOS program with the default NEARSTACK attribute generates the code that appears in Figure 1a, whereas a DOS program with FARSTACK defined would not need to adjust the stack pointer (Figure 1b). Under OS/2, DS is initialised so that it points to DGROUP and sets SS:SP to the starting address. The .EXIT directive generates a DOS interrupt (see Figure 2) or OS/2 system call to DosExit.

Program flow

Of course, very few programs execute sequentially from .STARTUP to .EXIT. MASM 6.0 contains several new high-level flow-control directives that generate code for loops and decisions depending on the status of the condition statement. The new

directives are .IF, .REPEAT, and .WHILE. The associated .BREAK and .CONTINUE directives are included, and may be used in conjunction with the C-like binary operators.

These directives give MASM 6.0 code a unique style. Their use can result in as much as a 70% reduction in source code length. Maintenance is simplified, and the likelihood that one piece of code can be reused for several target operating systems and processors without change is dramatically increased. Figure 2 shows a typical use of the decision directives and the code that is automatically generated by their use.

Most of the data types, structures and operators available in C have been duplicated in MASM 6.0. Whether or not this now makes MASM a hybrid is a matter for conjecture. It certainly makes MASM a very powerful tool. In the past, assembly language has never been the ideal medium for extensive multiple module programs - it lacks the control structures that ensure consistency between modules. Programmers have often used C as a 'wrapper' for a set of assembly routines to simplify maintenance and guard against the introduction of inconsistencies. This practice is rendered superfluous by MASM 6.0 features.

A new peripheral utility, H2INC, converts C header files into equivalent assembler include files. This is made possible by the introduction of new directives and data types that allow the translation of C headers to MASM without loss of data-type information. MASM 6.0 supports new signed types SBYTE, SWORD and SDWORD; floating-point data types REAL4, REAL8 and REAL10; and binary-coded decimals defined with TBYTE.

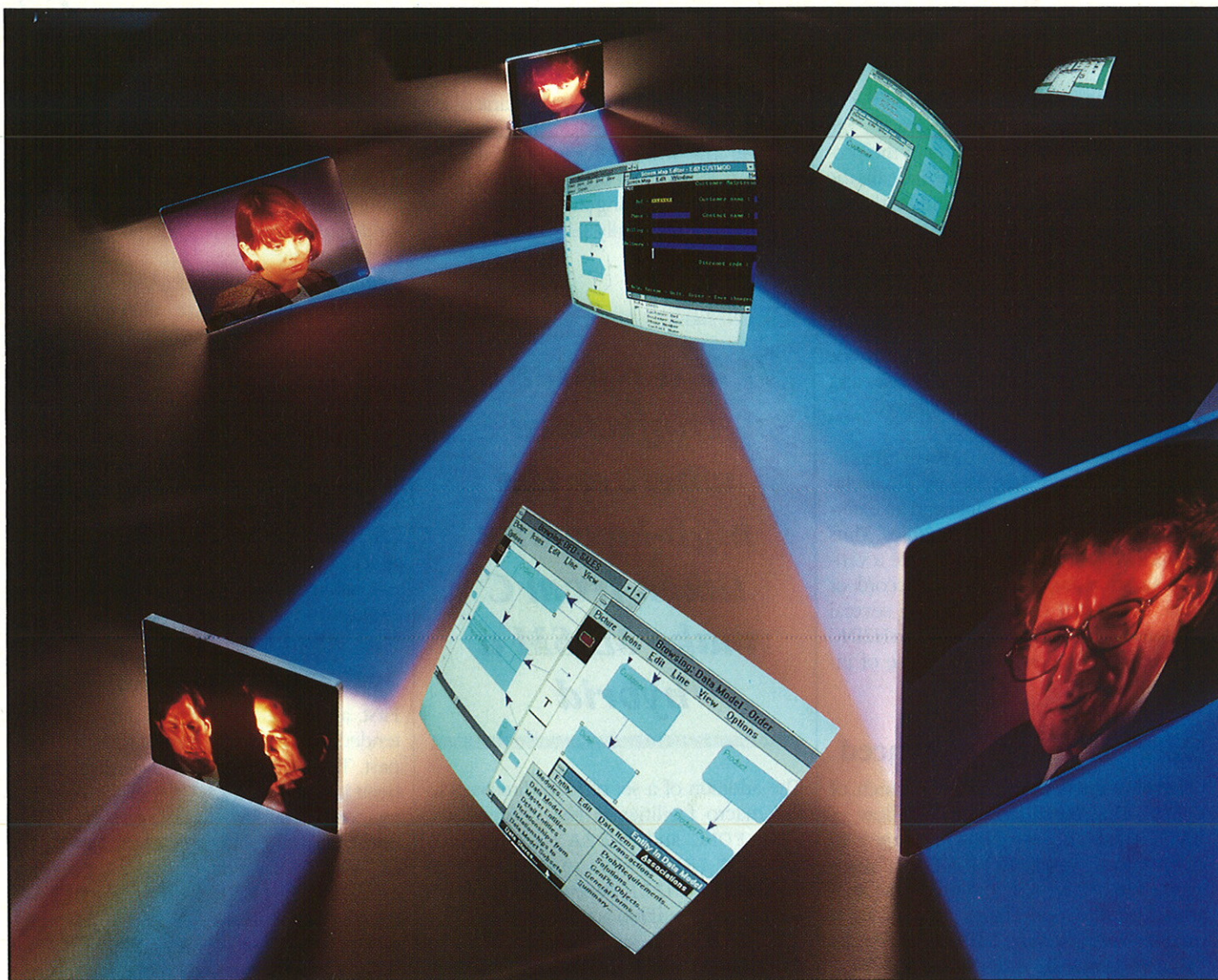
The PROTO directive defines procedure prototypes. Like C-language prototypes,

@Startup:	
mov dx, DGROUP	
mov ds, dx	
END @Startup	

Figure 1b - .STARTUP code in DOS mode with FARSTACK

```
@Exit:
mov al, exit_value
mov ah, 04ch
int 21h
END @Exit
```

Figure 2 - Code generated by .EXIT in DOS mode



LBMS Systems Engineer. Multi-user CASE to boost your team's productivity.

It's here! Practical CASE automation that takes team productivity to a new high...

LBMS Systems Engineer. Built on the LBMS pedigree, this multi-user windows-based tool offers leading-edge technology to support the rapid application techniques you want to use.

So why multi-user? Systems Engineer networks your team to provide concurrent and immediate access to shared information. The resulting accuracy and consistency are the keys to rapid progress in application development.

How does Windows help? Windows™ 3 is a breakthrough in performance and usability. Systems Engineer fully exploits this natural

desktop approach. As a result, multiple techniques can be used side-by-side to achieve the ideal decision-making context for quality designs. Systems Engineer transforms the PC into a powerful developer's workstation by integrating CASE with your choice of Windows™ tools (word processing, e-mail, project management etc.). For added flexibility, an OS/2™ version will be ready when you are.

As an IBM AD/Cycle™ vendor and partner to other leading industry suppliers, we offer an integrated set of CASE and methods solutions that can be matched perfectly to your development environment.

So that you gain the very best from our CASE tools and methods, we back you with the level of training and support that's appropriate to you. Our aim is to transfer the skills and experience of the best people in the business to your organisation.

Get LBMS integrated CASE solutions working with your team. Phone Elizabeth Baxter now on 071 636 4213 for a demonstration, more information or details of our CASE technology seminars.

Evelyn House 62 Oxford Street London W1N 9LF
Tel: 071-636 4213 Fax: 071-636 2708

LBMS

MASM prototypes inform the assembler of types and numbers of arguments so the assembler can check for errors in the procedure call. The INVOKE directive calls the procedure, converting arguments to the expected type and pushing parameters according to the correct calling conventions, as specified after the .MODEL directive (PASCAL, C, ...). If the VARARG keyword is used in place of a specific parameter, a variable number of arguments can be passed to the procedure. In this last case, parameter checking is clearly limited.

A new identifier EXTERNDEF is available. EXTERNDEF is treated as a public declaration in the defining module and an external declaration in accessing modules. Thus EXTERNDEF can be used to make a variable (a structure, array, union, record or basic data type) common among several modules. It was added together with H2INC to encourage the user to make use of include files.

Macro Language enhanced

Otherwise entitled 'the infernal march of GOTO'. That much berated bedfellow of BASIC (that also somehow managed to creep into C, Pascal etc) has finally managed to dupe Assembly Language, the final arbiter of 'All Good Things', into giving it quarter behind her hallowed portals (*Shurely jumps are GOTOS - Ed*). Well, there are some things in MASM 6.0 that seem almost out of place. That said, it is rumoured that there is a place for GOTO, and if that is true, then a macro is as good a place as any. The problem is, of course,

that GOTO is a much-abused means of escaping from dead-end logic, and often appears in code written after substantial liquid lunches. MASM 6.0 confines the GOTO to within macros, limiting the change of flow of control to a naturally self-contained portion of the code.

It is a matter of conjecture whether the high-level bits taken from C make MASM a hybrid

The addition of a set of loop directives to the macro facilities is less contentious. FOR, FORC and REPEAT are renamed forms of IRP, IRPC and REPT from MASM 5.1. WHILE repeats assembly as long as the condition is true.

By the use of the VARARG keyword, macros may now sport a variable number of arguments. Used together with the void pointer, you have the basis for a set of polymorphic functions, albeit rather stylised. This is possible because a void

pointer can take any argument, a dangerous but useful facility. Caveat programmer! The availability of a variable number of arguments is buttressed by the option to define a macro argument as either required or default using the REQ or := operator.

Numeric equates using EQU are still here, but the new TEXTEQU allows the programmer to assign a string equate. TEXTEQU can be used to assign the value calculated by a macro to a label, which has a mind-expanding potential for simple but powerful flow control. See Figure 4a for an example of some of the new macro commands. String handling capacity has also been added to EXITM - macro functions can now determine and return a text value. A set of pre-defined macros round off the extensions to the MASM 6.0 macro extensions, these are listed in Figure 4b.

Segments and OS support

First, a bug fix. MASM 5.1's faulty sign-extended addressing is now a thing of the past. The USE16/USE32 operands have been joined with FLAT to permit true flat memory model programming as available under the forthcoming OS/2 V2.0 and WIN-32. With the flat model attributes, the segment limit of 64 KB is done away with, and the theoretical limit of 4 GB (64 TB with memory mapping!) comes into play. OS/2 is not the only arena where the flat model may be used, as a number of 32-bit DOS extenders will support 32-bit-segment offsets, although normally limiting the segment size to around 16 MB. OS/2 2.0 limits it to 512 MB.

With the appropriate keyword in the .MODEL statement, the new .STARTUP directive generates the appropriate start-up code for either OS_OS/2 or OS_DOS, and the new .EXIT directive likewise generates the correct exit code. Here lies a potential problem for those intending to write flat model programs for a non-OS/2 platform. With the .386 or .486 directive preceding the .MODEL FLAT argument, only CS, DS, ES and SS are assumed under the FLAT supergroup, leaving FS and GS assumed to ERROR. This is because the 32-bit version of OS/2 reserves these registers for its own exclusive use. Likewise, the code generated for procedure initialisation and clean-up may be incorrect for a DOS extender. This is where the ability to define one's own setup/teardown code comes into its element, and the .OPTION PROLOGUE/EPILOGUE arguments save the day. All in all, the extended directives seem to have been well thought out. Most of the potential problems caused by high-level, automatic code-generating directives can

;asterisks mark the automatically generated code

```

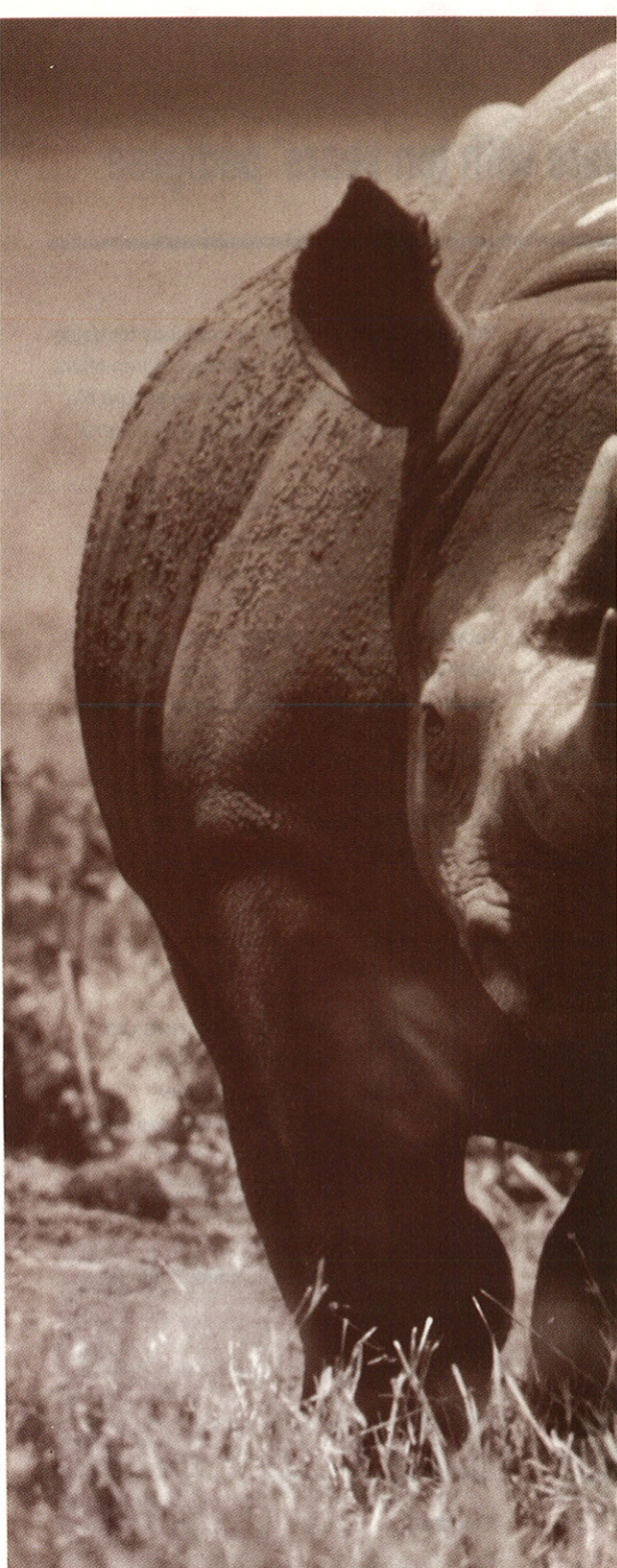
| |
| |
mov cx, 10
.REPEAT
* @C0001:
mov ah, 08h
int 21h
.BREAK .IF al == 13
* cmp al, 00dh
* je @C0003
.CONTINUE .IF (al < '0') || (al > '9')
* cmp al, '0'
* jb, @C0002
* cmp al, '9'
* ja @C0002

mov dl, al
mov ah, 02h
int 21h
.UNTILCXX
* @C0002:
* loop @C0001
* @C0003:
| |
| |

```

Figure 3 - The high-level decision directives

Software errors can be brutal... ...find them first with Evaluator



Testing Software properly means hours of boredom, and is very, very expensive. But letting users find errors can be as bruising as an encounter with our friend on the left.

Evaluator can help. A unique testing tool, Evaluator can automate the entire testing process, including regression, stress and acceptance testing. It can do this unattended, leaving staff free to perform more productive work.

Evaluator will test every kind of software including GUI applications such as Windows 3. Operating system independent you can test DOS, OS/2 or UNIX applications. If you can get your mainframe application to display on a PC it can test that too!

Evaluator is non-intrusive, which means that the machine you test your software on will be the same as the machine it gets used on; a very important quality consideration.

So if you don't want any unpleasant surprises from your software, call
The Products Group for full details on
0285 - 655888

ELVEREX

QA

Tick here for full details:

☐ Evaluator

☐ Product Catalogue

☐ Consultancy

☐ Training: OS/2, UNIX, Windows, C, Object Oriented, Networks.

CIRCLE NO. 984

NAME

TITLE

COMPANY

ADDRESS

TEL. NO.

Asynchronous Point-to-Point Communications or Telephone-polling Multiple Computers

XOREN COMMS SOFTWARE—Software with ten years' pedigree

Throughout the 1980s, Xoren Computing pioneered the development of inexpensive yet reliable communications software. Today, the XOREN IPL-11 range is renowned for enabling automated file transfer between any quantity and combination of the major minis, micros and PCs in configurations which will become more and more commonplace through the 1990s.

Leading with Automated Connectivity

Although many other communications packages can asynchronously link PCs together, very few besides IPL-11 will orchestrate *fully automated connectivity* between multiple PCs and hosts.

Automated operation with IPL-11 gives users the freedom to perform unattended data communications 24 hours a day. Computers with modems operating up to 9600 baud can be scheduled to dial up remote computers over ordinary telephone lines day or night to send and receive files. Any IPL-11 version can link and transfer files to any other IPL-11 version, regardless of the computers' operating system. Once the files have arrived safely, telephone lines are disengaged.

Leading with Customer Support

Key to the success of the IPL-11 range is an emphasis by Xoren on customer support.

Because communications is notorious for being a difficult application to implement, Xoren offers installation and support services to ensure the software can operate according to the *user's* requirements.

This principle has made IPL-11 communications software the obvious choice for UK and European companies. Because IPL-11 is a British product, developed and supported in the UK, Xoren is strategically placed to support customers when they need support most - during regular business hours.

Call or write to Xoren today - the most practical solution to automated data communications has always been right on your doorstep.

XOREN IPL-11 Developments over a Decade

1980:

Xoren launches IPL-11. Features include peer-to-peer file transfer under operator control, CRC error-checking on individual packets and queuing of file transfer commands.

Original versions introduced for RSX-11M and VMS for DEC PDP-11 and VAX.

1981:

Remote activation facility incorporated allowing file transfers when remote computers are unmanned.

RSTS/E, RT-11 and RSX-11M PLUS versions released for DEC PDP-11.

1982:

Terminal emulation facility introduced enabling the use of a terminal on a local computer as a terminal on a remote computer thereby allowing control of file transfer sessions from a single terminal.

TSX PLUS version released for DEC PDP-11.

1983:

Option to control file transfers from command files as an alternative to control from operator's keyboard.

P/OS version released for DEC Professional.

1984:

Mechanism built into the package protecting against "message bouncing" due to line noise when computers remain connected and the package is not in use.

MicroRSX and MicroRSTS versions released for DEC MicroPDP-11.

1985:

Commenced development of new portable versions written in the programme language 'C'.

1986:

First releases of new portable versions written in 'C' for PC-DOS, MS-DOS, UNIX, AIX and VMS.

1987:

Portable versions support simultaneous multiple links. IPL-11 wins ICP Million Dollar Award.

1988:

PC versions enhanced with improved terminal emulation including VT100 emulation, keyboard mapping and facilities to define function keys.

1989:

Comprehensive upgrade for unattended operation of multiple PC/host links supporting auto-dialling modems.

1990:

Release of MULTI-POLL range of polling software incorporating IPL-11 allowing PCs or other computers at multiple sites to be telephone-pollled by a central host computer system to transfer data to and from the central system.



XOREN COMPUTING LTD
28 MADDOX STREET
LONDON W1R 9PF
TELEPHONE 071-629 5932
FAX 071-629 5432

CIRCLE NO. 985

I am interested in Xoren Comms Software:

Name _____

Position _____

Company _____

Address _____

Postcode _____

.EXE 7/91


```
@ArgI MACRO index:REQ, arglist:VARARG
LOCAL count, retstr
count = 0
FOR arg, <arglist>
count = count + 1
IF count EQ index
retstr TEXT EQU <arg>
ENDIF
ENDM
EXITM retstr
ENDM
```

Figure 4a - Macro functions

be pre-empted by user-defined 'bolt-hole' macro directives.

The SEGMENT directive has been extended to allow user-control over how the linker combines segments from separate modules when building executable files. One may also use the AT combine type to force a segment address to a specific location. This is useful for defining structures or variables that correspond to specific far memory locations, such as the screen buffer, or the memory at offset 0x040:0000.

Windows

MASM 6 offers extensive Windows support. There is automatic setup and teardown of procedure stack frames for Windows applications. Most of the code in run-time Windows was written without the normal stack frames, using the undocumented C compiler switch -plmf. This may indicate that Microsoft is obsessed with making its code as teeth-grindingly tiny as possible, which isn't such a bad thing. More innovative is the LROFFSET directive. This generates a loader resolved OFFSET, allowing the Windows MMU to relocate code segments as needed:

```
INVOKE CreateRoutine,
LROFFSET MyFunc, 0
```

Taken together with the new high-level directives and much improved macro language, the assembly programmer now is in with a fair chance of writing a complete Windows program before retirement, incorporating a lot of the space and speed goodies previously available to punters. OS/2 support is extended as for Windows, with the addition of 32-bit offsets supporting the flat memory model of OS/2 V2.0 (and 386/486-specific extenders).

**Microsoft is
obsessed with
making its
code as
teeth-grindingly
tiny as possible**

Other bits

I've concentrated on MASM 6's new syntax in this article, so must rush through some other bits and pieces which you need to know. MASM is now shipped (and integrated with) the Programmer's Work Bench development environment, so MASM programmers can now do all the browsing, debugging (dear old CodeView is bundled, natch) and context-sensitive Helping previously only available to higher-level programmers. There is a DPMI version of the assembler executable which can handle extra-large source compilation. The off-line documentation has been completely re-

written, so there are no longer such discrepancies as existed in the MASM 5.1 Programmer's Guide. Among the several appendices, there is a full description of the MASM language in the Backus-Naur notation. The new assembler has a MASM 5.1 compatibility mode which allows the user to compile old code without a re-write.

MASM 6.0 gives the appearance of having been extensively beta-tested. However, only long-term use will show whether there have been any oversights in the development of this extremely advanced assembler.

Conclusion

With a price-tag similar to that of the MASM 5.1 it replaces, the new high-level directives and ultra-flexible macro extensions make MASM 6.0 the best buy at present on the market. The ability to contour the assembler's behaviour with user-defined code should be enough to keep the hackers happy. For users of the beta version of OS/2 2.0 and Windows programmers, MASM 6.0 is the only macro assembler that delivers the goods.

As I have indicated, there are aspects to the new assembler that would grace a structured, high-level language. However, at heart, MASM 6.0 is a considerably reworked and markedly improved replacement for MASM 5.1. If you want to buy a better MASM 5.1, this is it.

But to dismiss MASM 6.0 as a better MASM 5.1 would be to do Microsoft a great injustice, and to do you, the potential user, a great disservice. I believe that there is substance to Microsoft's claim that 'MASM 6.0 is not an evolutionary product that adds a few new features. Instead it is a revolutionary new version of MASM that reduces your coding time, simplifies code maintenance, and delivers exceptional performance.'

EXE

Don Binns apprenticed in digital electronics in the '70s, worked as a systems programmer during the '80s, thought better of it, and became a technical author and consultant in the '90s. Don Binns thinks he is 17, looks 45, but is actually only 33 years old. He is a confirmed optimist and still hopes to be paid for this article.

MASM 6 has an RRP of £105. Users of earlier versions of MASM can upgrade for £45. Microsoft sales department is on 0734 500741.

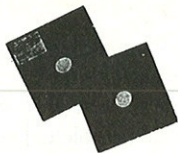
```
;=====
;@CatStr ;returns a concatenated string
;@InStr ;returns the position of one string within another
;@SizeStr ;returns the size of a string
;@SubStr ;returns a substring
;=====
;* @RestoreRegs - Macro to generate a pop instruction
;* for registers saved by the SaveRegs macro.
;* Restores the group of registers most recently
;* pushed -- that is, RestoreRegs pops the group of
;* registers saved by the last use of SaveRegs.
;
@RestoreRegs MACRO
LOCAL regs, numloc
numloc InStr %1, regpushed, <!>> ;Find end of list
regs SubStr regpushed, %1, %numloc + 1;Get list of registers
pushed SubStr regpushed, %numloc ;Remove list from pushed
FOR reg, @ArgRev( regs ) ;Pop each register
pop reg
ENDM
ENDM
;=====
```

Figure 4b - Pre-defined macros

BRENT

Communication and Telemetry Consultants

"MaxPro – the ultimate in hardware for software protection"



The MaxPro Data key fitted to the Parallel Port is totally transparent to the end user.

Software customers make as many backups as are required but of course only the machine using the individually coded MaxPro key will run the program.

BEWARE THE PIRATE'S PATCH

High security encryption is created by the Simple-to-Use Menu-driven MaxPro attachment software. Other data keys may be patched out in the software rendering it totally vulnerable. MAXPRO protected files cannot be patched. Choose whether or not to shut down the program if tampering occurs.

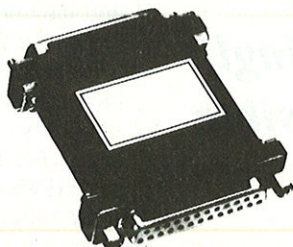
Optional limited life and Usage information collection facilities are provided.

No source or object code modules needed.

The MaxPro system works on IBM; PC; XT; AT Compatibles and PS2.

For additional information contact us at:

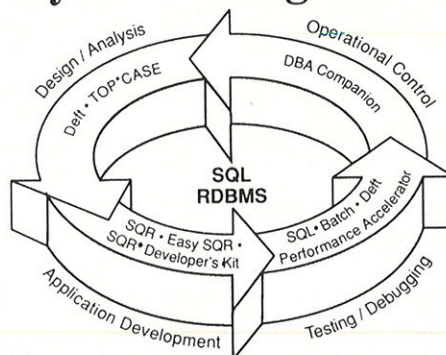
Brent Communications
Unit 2
Dragon Industrial Estate
Harrogate HG1 5DN



Tel: (0423) 566972
Fax: (0423) 501442

CIRCLE NO. 986

SQL Solutions "THE" Systems Integrators



SQL Solutions is the first single source vendor of professional services and SQL productivity tools to provide complete SQL Systems Integration.

We offer:

- * RDBMS independent Productivity tools to address each phase of the applications development lifecycle.
- * Unparalleled expertise in the design and implementation of applications in all major RDBMS's.
- * Gateway products to enable relational databases to co-exist.

SQL Solutions (UK) Ltd
Kennet House, Waterside Park
Sweetwell Road
Bracknell, Berks, RG12 1HH

Tel: 0344 714053 Fax: 0344 714056

CIRCLE NO. 987

The Editorial Team of .EXE^[Will] wishes to make **A CALL FOR PAPERS** ^[A request for articles]

Forthcoming Editorial Thematic Issues:

^[Topics coming up] October - Mostly UNIX
November - GUI's
December - Back to the Future

Other ideas always welcome

^[ie the lazy s*xd can't think of them himself]

All published work generously rewarded ^[Ha!]

Send away for a stylish, individually bound copy
of our Contributors' Notes

^[A few xeroxed sheets inexpertly stapled together]

Contact The Editor, .EXE Magazine, 10 Barley Mow Passage, London W4 4PH

3D Computer Graphics - The Floating Horizon Algorithm

In the second part of his 3D graphics series, Graeme Webster builds on the basic transformations to give a version of the popular Floating Horizon algorithm for displaying surfaces.

In the May 1991 issue of .EXE, a base camp was established in preparation for an assault on 3D computer graphics. Of necessity this involved some fairly heavy algebra and the code that went with it. Carrying on with the mountaineering metaphor, even attaining some of the lesser peaks, such as flat shaded solids is going to involve more hard uphill work. The giddy heights of ray-tracing and radiosity techniques for photorealism still further effort. So, by way of light relief, we will spend this article pottering about in the foot hills, practising using our new tools to produce a definitive version of that old chestnut, the floating horizon algorithm.

The floating horizon algorithm is most often used to display a three dimensional representation of surface functions of the form $z=Z(x,y)$

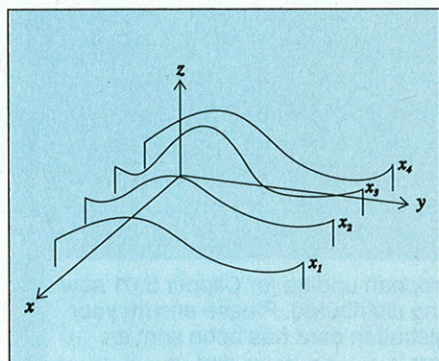


Figure 1 - Curves on parallel planes of constant x -value

from which the hidden lines have been removed. Functions of this form arise in

The fundamental idea behind the algorithm is to convert the three-dimension problem

into a set of overlapping two-dimensional ones by slicing through the surface with a number of planes parallel to one of the coördinate axes. This is shown in Figure 1, where the planes have constant values of x . The function Z has now been reduced to a set of curves defined in these parallel planes and the problem is to display these curves so that only the parts shown are those visible from the selected view point. The images generated are often quite pleasing in themselves and, as simple implementations

are rather easy, these have sometimes found their way into elementary graphics programming books. However, the floating horizon is a genuine hidden line elimination problem which, as with all such problems, has a number of subtleties and tricky bits. If these are ignored, things can go wrong.

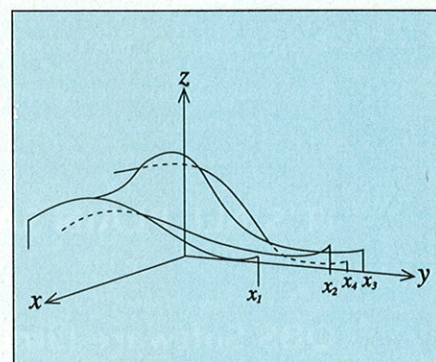


Figure 2 - Curves Partially Hidden

many areas of science and engineering. Contour maps can be elevated into topographical models and our old friend the Mandelbrot set turned into a mountain plateau by trading the colour variations of the usual 2D representations into z -heights. The one necessary condition which they must satisfy is that they are single valued in z , ie for any pair of coördinates (x,y) , Z has just one value. That is reasonable enough for topographical models, but cuts out some familiar mathematical objects such as balls and tori ('doughnuts') which are two valued, having a top and a bottom. (Of course, if you *know* that the function is two valued, you can fudge things to allow for it; but the general case would have to cope with a ball made by screwing up a sheet of paper which, in principle, has an arbitrary number of values of z for particular x s and y s). As we will see later, even this single valued condition is not sufficient when the function is viewed in perspective.



BLINKER™ TAKES CLIPPER™ OUT OF THIS WORLD

Blinker eliminates the need for complex overlay structures for both Clipper Summer '87 and 5.0

- * Significantly reduces run-time memory requirements
- * Accelerates your development cycle 3-10 times faster than any other current linker and even faster when linking incrementally
- * Dynamically overlays C and Assembler code such as Extend.lib, dGE, FUNCKy, other Clipper Add-ons and your own 'C' code
- * Optimises your application with advanced profiling functions
- * Includes memory defragmentation, 'burning-in' of Clipper environmental variables/serial numbers and creation of demonstration versions

IT'S ALL PLAIN SAILING WITH BLINKER

European sales and support:

QBS Software Limited

10 Barley Mow Passage, Chiswick, London W4 4PH
Telephone 081-994 4842. Fax 081-994 3441. BBS 081-747 1979

All trademarks recognised

What the experts say

Here are just some comments from the 5000 happy users to date

"It's blinking fast!"

"Congratulations – we gained over 100k"

"It's so good, I went through all my programs and re-compiled/linked just for the pleasure of seeing all the free memory"

"It's fast! Handles overlays like a dream"

"Amazing – No more messing about with overlay structures!"

"Brilliant!... But my coffee breaks have been shortened!"

"Actually exceeded our most optimistic performance hopes! (Just for once)"

European update for Clipper 5.01 now being distributed. Please ensure your registration card has been sent to:

**Blink Inc Limited, TCL Centre,
Cwmbran NP44 3XR**

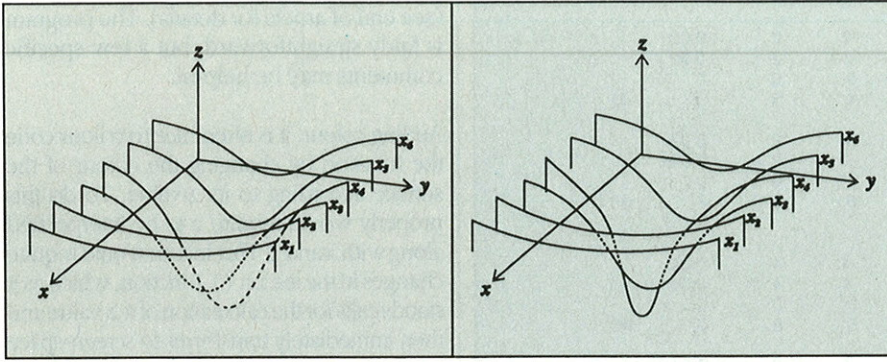


Figure 3 - Curves which project under the surface

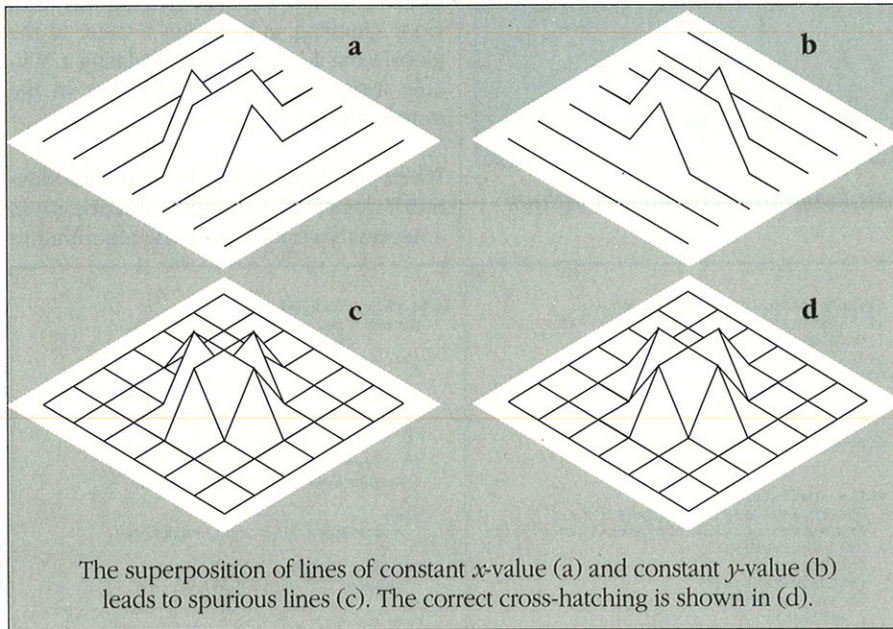


Figure 4 - Dealing with Cross-hatching

The algorithm first sorts the slices into increasing distance from the eye. Beginning with the plane closest to the eye, points along the curve of the slice are calculated and projected onto the viewing plane as a point ($x_{\text{screen}}, y_{\text{screen}}$). The simplest form of the hidden line algorithm is then:

'If y_{screen} is greater than the screen y -value for any previous curve a the same x -value, the point is visible and should be plotted; otherwise, it is invisible.'

Figure 2 shows how this works. Having changed the view point from that of Figure 1, the dashed portions of the lines are now hidden. The maxima of the curves form a horizon below which things are invisible. As more curves are added this horizon 'floats' upwards - whence the name of the technique.

Implementation is straightforward. An array is created of a size equal to the horizontal resolution of the screen. The array's elements are initialised to some large negative value before adding the curves.

The first improvement is to allow for curves which dip down below the lowest curve which, with a single floating horizon, would be nipped off (Figure 3a). Adding a second horizon array which starts at large positive values and floats down recording the lowest points of curves solves this problem (Figure 3b), the algorithm becoming:

'If y_{screen} is greater than the maximum screen y -value or less than the minimum screen y -value for any previous curve a the same x -value, the point is visible and should be plotted; otherwise, it is invisible.'

As stated, the algorithm assumes that we have a screen y -value for each screen x -value along the curves. In practice this will not be the case, and it will be necessary to do some interpolation. Clearly it is not sufficient to calculate an interpolation and push it straight into the horizon arrays on the basis of the visibility of its end points, since all sorts of undulations might occur in the intervening space. Some implementations endeavour to calculate where the interpolation cuts the horizons and act

accordingly. On the whole it is simpler - and probably as efficient - to work along x -value by x -value, checking as you go.

Very narrow spikes can also cause a problem if the sampling is too sparse. This is a classical aliasing problem (like, eg, jagged lines in computer graphics or wagon wheels apparently rotating the wrong way in Westerns). The way to overcome these problems is to increase the sampling rate; in this case, to calculate points on the function closer together where there are spikes.

Often it is desired to plot the surface with curves of both constant x and constant y , thereby giving it a cross-hatched appearance. At first thought, it would seem that in order to do this all that is necessary is to apply the basic algorithm twice, once for x -slices and a second time for y -slices. Figure 4 shows that this is not the case; a simple superposition gives rise to spurious lines. Achieving a correctly cross-hatched surface requires a modification to the algorithm. First we determine which set of slices is the more nearly horizontal when projected onto the screen. The more nearly horizontal slices are processed in the usual way, working away from the eye. However, after each of these slices is processed, the short segments of the orthogonal slices lying between it and the next parallel slice are processed before the next nearly horizontal slice is dealt with. The result is that, at intermediate stages the drawing of the surface has a comblike edge. Finally, the same pair of floating horizons are used for both sets of slices.

The Program

These ideas are incorporated into the listing which accompanies this article. The code makes use of the Graphics Library previously published in .EXE Magazine. The program can be keyed in then compiled and linked with that library, alternatively it is available on the usual basis from .EXE

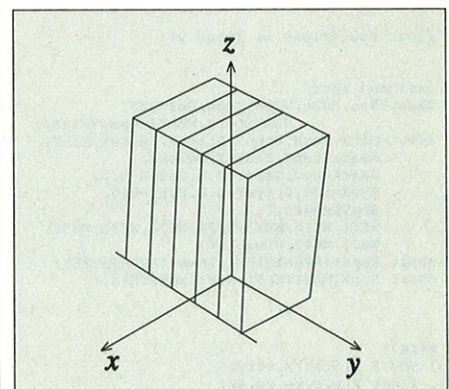


Figure 5 - Perspective problems with single valued functions

Sombrero:					
1.	x-range	-2	2	0.02	
	y-range	-1	2	0.02	
	centre of gaze	0	0	0	
	eye position and viewing angle	4	5	-2	45
2.	x-range	-2	2	0.02	
	y-range	-1	2	0.02	
	centre of gaze	0	0	0	
	eye position and viewing angle	6	5	4	36
Four spikes:					
1.	x-range	-4	4	0.04	
	y-range	-4	4	0.04	
	centre of gaze	0	0	0	
	eye position and viewing angle	5	6	7	45
2.	x-range	-4	4	0.04	
	y-range	-4	4	0.04	
	centre of gaze	0	0	-2	
	eye position and viewing angle	8	16	-20	30
Mandelbrot mountain:					
	x-range	-2	2	0.02	
	y-range	-2	2	0.02	
	centre of gaze	0	0	1	
	eye position and viewing angle	30	-40	50	4

Figure 6 - Suggested ranges of x-, y- and viewing parameters values

(see end of article for details). The program is fairly straightforward, but a few specific comments may be helpful.

Adding colour: it is often nice to colour code the function by changing the colour of the surface according to its z-value. To do this properly would require z to be interpolated along with x and y. That in turn would require changes in the main() function, which as it stands calls for the calculation of a z-value and then immediately transforms to screen-space where the actual interpolation is carried out. A 'cheap' way, which is reasonably satisfactory for gently undulating functions with close sampling, is to assign a value to the global variable Col when calculating z. Suitable assignments are pointed out in the sample functions for plotting.

When plotting goes wrong: in the introduction it was stated that single valuedness was a necessary condition for the algorithm to

<pre>// floathor.c floating horizon program #include <ctype.h> #include <graph.h> #include <math.h> #include <search.h> #include <stdio.h> #include <stdlib.h> #include <video256.h> #define TRUE 1 #define FALSE 0 #define BIG 32767 #define ARRAYSIZE 1024 #define MAXSTEPS 1024 #define PI 3.14159 #define PI180 0.0174533 short GetData(void); void CalcViewCoeffs(void); void DoPerspective(float x, float y, float z, short *scnx, short *scny); void GetCentreOfGaze(void); void GetEyePoint(void); void Horizons(short x0, short y0, short x1, short y1); void InitArrays(void); void PixelAndHorizons(short *type, short *HRes, short *VRes); void SwapF(float *pa, float *pb); void SwapI(short *pa, short *pb); void TestOrientation(void); void TransformWorldToEye(float xw, float yw, float zw, float *xe, float *ye, float *ze); float Func(float x, float y); unsigned short HRes, VRes, HCent, VCent, Type, Col=255, NumXSteps, NumYSteps, YFirst; float MinX, MaxX, StepX, StartX, DeltaX, MinY, MaxY, StepY, StartY, DeltaY, GazeX=0.0, GazeY=0.0, GazeZ=0.0, EyeX=300.0, EyeY=0.0, EyeZ=0.0, AngView=45.0, WE11, WE12, WE13, WE21, WE22, WE23, WE32, WE33, WE43, ViewDist; short Upper[ARRAYSIZE], Lower[ARRAYSIZE]; short ScrX[MAXSTEPS], ScrY[MAXSTEPS]; main() { short i, j, scrx, scry; float x, y, z, xe, ye, ze;</pre>	<pre>SetUpGraphics(&Type, &HRes, &VRes); if (InitGraphics256(Type, HRes)==0) { EndGraphics256(); printf("Invalid video adaptor or\ resolution\n"); exit(1); } SetDefaultPalette256(1.6); HCent=HRes>>1; VCent=VRes>>1; while (GetData()!=0) { GetCentreOfGaze(); GetEyePoint(); FilledRectangle256(0,0,HRes-1,VRes-1,0); CalcViewCoeffs(); InitArrays(); TestOrientation(); if (YFirst==TRUE) { x=StartX; for (i=0;i<=NumXSteps;i++) { y=StartY; for (j=0;j<=NumYSteps;j++) { z=Func(x,y); TransformWorldToEye(x-GazeX, y-GazeY, z-GazeZ, &xe, &ye, &ze); DoPerspective(xe, ye, ze, &scrx, &scry); if (i!=0) Hori- zons(ScrX[j], ScrY[j], scrx, scry); ScrX[j]=scrx; ScrY[j]=scry; if (j!=0) Horizons(ScrX[j-1], ScrY[j-1], scrx, scry); y+=DeltaY; } } else { y=StartY; for (i=0;i<=NumYSteps;i++) { x=StartX; for (j=0;j<=NumXSteps;j++) { z=Func(x,y); TransformWorldToEye(x-GazeX, y-GazeY, z-GazeZ, &xe, &ye, &ze); DoPerspective(xe, ye, ze, &scrx, &scry); if (i!=0) Horizons(ScrX[j], ScrY[j], scrx, scry); ScrX[j]=scrx; ScrY[j]=scry; if (j!=0) Horizons(ScrX[j-1], ScrY[j-1], scrx, scry); x+=DeltaX; } } y+=DeltaY; } } Rectangle256(0,0,HRes-1,VRes-1,252); getch(); } EndGraphics256();</pre>	<pre>void CalcViewCoeffs(void) { double rxy, rxyz, costh, sinth, cosph, sinph; char buffer[81]; rxy=sqrt(EyeX*EyeX+EyeY*EyeY); rxyz=sqrt(EyeX*EyeX+EyeY*EyeY+EyeZ*EyeZ); if (rxy==0.0) { costh=1.0; sinth=0.0; } else { costh=EyeX/rxy; sinth=EyeY/rxy; } if (rxyz==0.0) { cosph=1.0; sinph=0.0; } else { cosph=EyeZ/rxyz; sinph=rxy/rxyz; } // Coefficients of world- to eye-space // transformation equations WE11=-sinth; WE12=-cosph*costh; WE13=-sinph*costh; WE21=costh; WE22=-cosph*sinth; WE23=-sinph*sinth; WE32=sinph; WE33=-cosph; WE43= rxyz; // Viewing distance ViewDist=0.5*HRes/ tan(0.5*PI180*AngView); } void DoPerspective(float x, float y, float z, short *scnx, short *scny) { *scnx=ViewDist*x/z; *scny=ViewDist*y/z; } void GetCentreOfGaze(void) { unsigned char response[81]; Input256("Centre of gaze ", 0, 64, 252, 0, response); if (strlen(response)>0) sscanf(response, "%f %f %f", &GazeX, &GazeY, &GazeZ); } short GetData(void) { unsigned char response[81]; static int firstgo=TRUE; FilledRectangle256(0,0,HRes-1,15,0); if (firstgo) firstgo=FALSE; else { Input256("Another view? ", 0, 0, 252, 0, response); if ((response[0]!='n') </pre>
---	--	--

Figure 7 - Floating horizon program

Bring Graphs and Charts to your Windows Applications Licence-Free with Graphics Server™

Introducing the **Graphics Server Software Development Kit (SDK)** from Bits Per Second. The most cost-effective, time-efficient way to integrate full-colour graphs and charts into your Windows applications. Here's why.

Client-Server Model

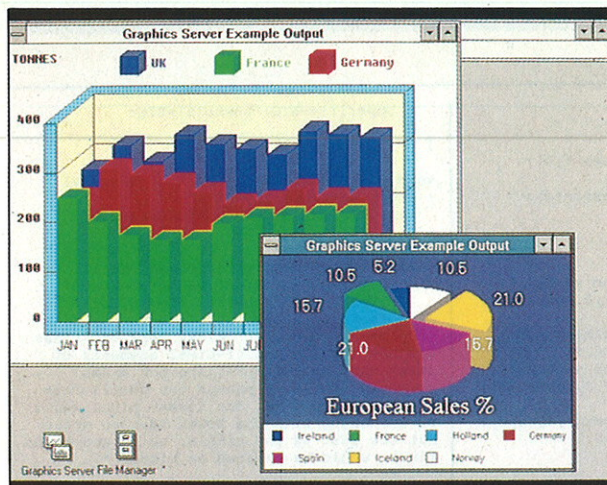
Graphics Server is a Windows server process which responds to multiple, concurrent requests for interactive graphing and charting functions from client applications. It enables graphs and charts to be incorporated into any Windows application having a Dynamic Link Library (DLL) or Dynamic Data Exchange (DDE) capability. This includes C programs, Superbase, SQL Windows, ToolBook, Word for Windows and many others. In fact, virtually all Windows applications!

Multiple Choice

Graphics Server lets you choose from 16 graph types, including 2-D and 3-D bar (simple, stacked and clustered), 2-D and 3-D pie, area, Gantt, scatter, polar, log-log/log-linear, line, high-low-close, X-Y and polar, plus a variety of statistical functions.

Ease of Programming

Graphics Server, through its multi-level interface, provides the level of control you want. In as few as **eight** calls you can open a new graph window, define titles, legends and labels, transfer data, draw a chart, and output the chart to the clipboard or printer. Graphics Server takes care of all the details; scaling the graph to match the data, matching fonts, text size and position to the graph, even resizing and repainting the graph window when necessary! You'll save development time. And money.



Alternatively, if you want full control yourself, Graphics Server provides over a **hundred** functions to help you design, create and manage virtually any kind of graphically drawn image. Multiple graphs, multiple windows, user interaction via hot regions. All these and more!

Full Support

The Graphics Server SDK gives you all the tools, documentation and examples you'll need to make immediate and effective use of Graphics Server. Within hours your applications will be transformed by the inclusion of presentation-quality graphics.

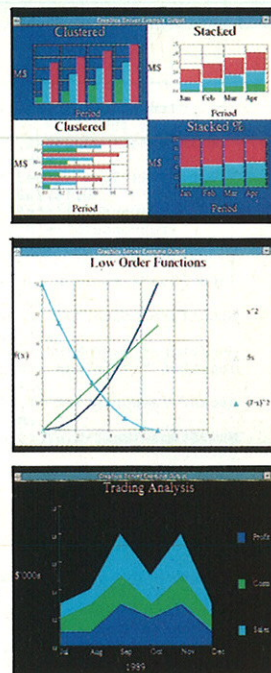
Telephone/fax support and a bulletin board service are provided for all registered users, and for power users requiring an even higher level of service and security there's a special VIP Support Programme.

No Licences!

The Graphics Server SDK costs £275 + VAT. No further licence fees to pay.

To place an order, request a demo disk, or obtain additional information, call . . .

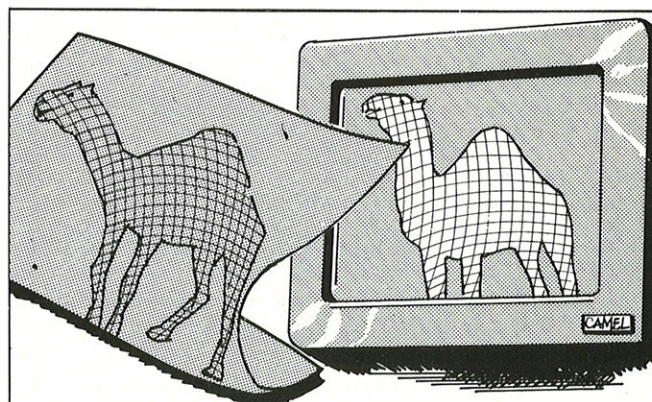
All product names are trademarks of their respective holders.



**BITS PER
SECOND**

Bits Per Second Ltd
14 Regent Hill
Brighton BN1 3ED
Tel (0273) 727119
Fax (0273) 731925

CIRCLE NO. 990



PLOT THE DIFFERENCE PLOTVIEW

Is the simple, low cost alternative to paper plots. Able to preview plot images on a graphics screen within seconds.



PLOTVIEW allows paperless plotting and replotting - avoiding delays, improving efficiency.

PLOTVIEW

Has up to 25 separate plot files previewable simultaneously on screen and a zoom function which allows any part of the plot to be inspected in detail.



Easy to integrate, PLOTVIEW is ideal for system builders and can be customised to suit their individual needs.



Phone now for more details about PLOTVIEW: the versatile tool that can really make a difference to your productivity.

CAMEL SERVICES LTD.
Telephone OXFORD (0865) 512678

CIRCLE NO. 991

CopyControl

THE NEW GENERATION OF COPY PROTECTION

- NO** User hassle
- NO** Back-up problems
- NO** Hardware add-ons or special disks needed
- NO** Changes to source code required
- YES** CopyControl beats ALL bit-copier Programs
- YES** Floppy disks, hard disks and networks supported
- YES** CopyControl is totally transparent to the user
- YES** You can limit program use by no. of copies, no. of executions or date
- YES** CopyControl works on all IBM compatibles
- YES** Free demo disk available

For Further Information Phone or Write to:

microcosm

Microcosm Limited,
17 Cranbrook Road, Bristol BS6 7BL
Telephone: 0272-441230 Fax: 0272-427295

CIRCLE NO. 992


```

(response[0]!='N'))
return (0);
FilledRectangle256(0,0,
    HRes-1,VRes-1,0);
Input256("Same range? ",
    0,16,252,0,response);
if ((response[0]!='n') &&
    (response[0]!='N'))
    return (1);
}
Input256("X-minimum x-maximum x-step ",
    0,32,252,0,response);
sscanf(response,"%f %f %f",
    &MinX,&MaxX,&StepX);
NumXSteps=(MaxX-MinX)/StepX+0.5;
Input256("Y-minimum y-maximum y-step ",
    0,48,252,0,response);
sscanf(response,"%f %f %f",
    &MinY,&MaxY,&StepY);
NumYSteps=(MaxY-MinY)/StepY+0.5;
return(1);
}

void GetEyePoint(void)
{ unsigned char response[81];

    Input256("Eye point and angle of view ",
        0,80,252,0,response);
    if (strlen(response)>0)
        sscanf(response,"%f %f %f %f",
            &EyeX,&EyeY,&EyeZ,&AngView);
    FilledRectangle256(0,0,HRes-1,15,0);
}

void Horizons(short x0,short y0,
    short x1,short y1)
{ short d,dx,dy;
  short Aincr,Bincr,xincr,yincr;
  short x,y;

  if (abs(x1-x0)>abs(y1-y0))
  { // for lines with slope between
    // -1 and 1
    if (x0>x1) // force x0<x1
        { SwapI(&x0,&x1); SwapI(&y0,&y1);
        }
    if (y1>y0) // determine increment for y
        yincr=1;
    else
        yincr=-1;
    dx=x1-x0; // initialise constants
    dy=abs(y1-y0);
    d=2*dy-dx;
    Aincr=2*(dy-dx);
    Bincr=2*dy;
    x=x0; // initial x and y
    y=y0;
    PixelAndHorizons(x,y);
    for (x=x0+1;x<=x1;x++)
    { if (d>=0)
        { y+=yincr;
          d+=Aincr; // set pixel A
        }
        else
          d+=Bincr; // set pixel B
        PixelAndHorizons(x,y);
      }
    }
  else
  { // for lines with slope less than -1
    // and greater than 1
    if (y0>y1) // force y0<y1
        { SwapI(&x0,&x1); SwapI(&y0,&y1);
        }
    if (x1>x0) // determine increment for x
        xincr=1;
    else
        xincr=-1;
    dy=y1-y0; // initialize constants
    dx=abs(x1-x0);
    d=2*dx-dy;
    Aincr=2*(dx-dy);
    Bincr=2*dx;
    x=x0; // initial x and y
    y=y0;
    PixelAndHorizons(x,y);
    for (y=y0+1;y<=y1;y++)
    { if (d>=0)
        { x+=xincr; // set pixel A
          d+=Aincr;
        }
        else // set pixel B
          d+=Bincr;
        PixelAndHorizons(x,y);
      }
    }
}

void InitArrays(void)
{ short i;

  for (i=0;i<ARRAYSIZE;i++)
    { Upper[i]=-BIG; Lower[i]=BIG;
    }
}

void PixelAndHorizons(short x, short y)
{ short centrex;
  static short oldx=BIG,oldlower=BIG;

  centrex=HCen+x;
  if ((0<=centrex) && (centrex<HRes))
  { if ((centrex==oldx) && (y<oldlower))
      SetPixel256(centrex,VCen-y,Col);
    /* Without this subtlety, and the setting
    of oldx and oldlower below, vertical lines
    and the vertical multi-pixel segments of
    lines with abs(slope)>1 are not drawn
    properly. Horizons passes out pixels from
    the lowest upwards. The lowest pixel would
    be set along with the Lower horizon array
    but the rest of the pixels, being above the
    first, would be treated as hidden */

    if (y<Lower[centrex])
    { oldlower=Lower[centrex];
      oldx=centrex;
      Lower[centrex]=y;
      SetPixel256(centrex,VCen-y,Col);
    }
    if (Upper[centrex]<y)
    { Upper[centrex]=y;
      SetPixel256(centrex,VCen-y,Col);
    }
  }
}

void SetUpGraphics(short *type,
    short *HRes,
    short *VRes)
{ printf("Type of video adaptor:\n");
  printf("0Video-7,1Paradise,2ATI\
  Wonder,3Tecmar,4SOTA,5Orchid,6EIZO ");
  scanf("%d",type);
  printf("Pixels per line ");
  scanf("%d",HRes); *VRes=3*(HRes)/4;
  if (InitGraphics256(*type, *HRes)==0)
  { EndGraphics256();
    printf("Invalid mode or resolution\n");
    exit(1);
  }
}

void SwapF(float *pa, float *pb)
{ float t;

  t=*pa; *pa=*pb; *pb=t;
}

void SwapI(short *pa, short *pb)
{ short t;

  t=*pa; *pa=*pb; *pb=t;
}

void TestOrientation(void)
{ float xemin,yemin,zemin;
  xemax,yemax,zemax;
  short x,y,xdx,xdy,ydx,ydy;

  TransformWorldToEye(MinX,MinY,0,
    &xemin,&yemin,&zemin);
  TransformWorldToEye(MaxX,MinY,0,
    &xemax,&yemax,&zemax);
  if (zemax<zemin) // MaxX closer than MinX
  { StartX=MaxX; DeltaX=-StepX;
  }
  else // MinX closer than MaxX
  { StartX=MinX; DeltaX= StepX;
  }
  DoPerspective(xemin,yemin,zemin,
    &xdx,&xdy);
  DoPerspective(xemax,yemax,zemax,&x,&y);
  xdx=-x; xdy=-y;
  TransformWorldToEye(MinX,MinY,0,
    &xemin,&yemin,&zemin);
  TransformWorldToEye(MinX,MaxY,0,
    &xemax,&yemax,&zemax);
  if (zemax<zemin) // MaxY closer than MinY
  { StartY=MaxY; DeltaY=-StepY;
  }
  else // MinY closer than MaxY
  { StartY=MinY; DeltaY= StepY;
  }
  DoPerspec-
  tive(xemin,yemin,zemin,&ydx,&ydy);
  DoPerspective(xemax,yemax,zemax,&x,&y);

  ydx=-x; ydy=-y;

  /* if abs(slope-y)<abs(slope-x), ie
  abs(ydy/ydx)<abs(xdy/xdx) or
  abs(ydy*xdx)<abs(xdy*ydx) draw in
  y-direction first then add the x-going bits
  */
}

when eye is in the plane of the axes this
breaks down so then use the test in the eye
location */

if (EyeZ!=0)
{ if (abs(ydy*xdx)<=fabs(xdy*ydx))
    YFirst=TRUE;
  else
    YFirst=FALSE;
}
else
{ if (abs(EyeX)>abs(EyeY))
    YFirst=TRUE;
  else
    YFirst=FALSE;
}
}

void TransformWorldToEye(float xw,
    float yw,
    float zw,
    float *xe,
    float *ye,
    float *ze)
{ *xe=WE11*xw+WE21*yw;
  *ye=WE12*xw+WE22*yw+WE32*zw;
  *ze=WE13*xw+WE23*yw+WE33*zw+WE43;
}

// ***** Function to draw *****

// rename one of the following as Func
// before compiling!!!!!!

// ***** Function to draw *****
float Func1(float x, float y)
// Sombrero
{ float r2,z;

  r2=x*x+y*y;
  z=0.2*sin(x)*cos(y)+
    1.5*cos(2.0*r2)*exp(-r2);
  // colour-coding z height
  Col=fabs(150*z);
  return(z);
}

// ***** Function to draw *****

// ***** Function to draw *****
float Func2(float x, float y)
// Four spikes
{ float r12,r22,r32,r42,r52,z;
  static float
    a1= 2.0, b1= 2.0, c1= 1.0,
    a2= 2.0, b2=-2.0, c2= 1.0,
    a3=-2.0, b3= 2.0, c3= 1.0,
    a4=-2.0, b4= 2.0, c4= 1.0,
    a5= 0.0, b5= 0.0, c5=-4.0;

  r12=(x-a1)*(x-a1)+(y-b1)*(y-b1);
  if (r12<0.5) r12=0.5;
  r22=(x-a2)*(x-a2)+(y-b2)*(y-b2);
  if (r22<0.5) r22=0.5;
  r32=(x-a3)*(x-a3)+(y-b3)*(y-b3);
  if (r32<0.5) r32=0.5;
  r42=(x-a4)*(x-a4)+(y-b4)*(y-b4);
  if (r42<0.5) r42=0.5;
  r52=(x-a5)*(x-a5)+(y-b5)*(y-b5);
  if (r52<0.5) r52=0.5;

  z=c1/r12+c2/r22+c3/r32+c4/r42+c5/r52;
  // colour-coding z height
  Col=fabs(120*z);
  return(z);
}

// ***** Function to draw *****

// ***** Function to draw *****
float Func3(float x, float y)
// The Mandelbrot mountain
{ float zr,zi,zr2,zi2,tr,z;
  int n;

  n=0; zi2=zr=zi=zr2=0.0;
  do
  { tr=zr2-zi2+x; zi=2.0*tr*zi+y; zr=tr;
    zr2=zr*tr; zi2=zi*zi; n++;
  } while (((zr2+zi2)<4.0)&&(n<255));
  // alternatives for z below
  // z=0.01*((float)n);
  z=0.40*log((float)n);
  // alternatives for z above
  // colour-coding z height
  Col=fabs(100*z);
  return(z);
}

// ***** Function to draw *****

```

Figure 7 - Floating horizon program (Continued)

Everything You Ever Wanted in UNIX for only £99.95*

By the end of this year there will be more people using UNIX™ than any other operating system in the world. But, for many of us UNIX is way beyond our pocket — it might as well be on another planet.

But now Coherent™ gives you everything you ever wanted from UNIX, but for less. You might find that hard to believe, but just consider this.

Coherent is a virtual clone of UNIX, and was developed independently by the Mark Williams Company. That means you don't have to pay hundreds of pounds per copy in licensing fees.

Smaller, Faster and Better!

What's more, Coherent embraces the original principle of UNIX — small is beautiful.

This leads to the fact that Coherent has a whole host of cost and performance advantages. So read on, because there's a lot more to Coherent than just its keen price.

For one thing, Coherent gives you UNIX capabilities on a machine you can actually afford. Needing only 10 megabytes of disk space, Coherent can reside with DOS. So you can keep all your current software and move up to Coherent at your own speed. You can also run it faster, learn it quicker and get better overall performance. All because Coherent is small. Sounds to good to be true, doesn't it?

But small wouldn't be so great if it didn't do the job it was meant to do.

Everything Unix Was Meant To Do and More!

Like the original UNIX, Coherent is a powerful multi-user, multi-tasking development system. With a complete UNIX-compatible kernel which opens up the vast world of UNIX software, including over a gigabyte of public domain software.

Everything You Need for Less!

Coherent also comes with Lex & Yacc, UUCP, a C compiler and a full set of almost 200 UNIX commands for text processing, program development, administration and maintenance.

The latest release includes a UUCP Bulletin Board System, RAM disk support and much, much more.

	Coherent for IBM-PC/AT 286 or 386 machines.	Santa Cruz XENIX 286, Version 2.3.2
No. of Manuals	1	8
No. of Disks	4	21
Kernel Size	64 Kb	198 Kb
Install Time	20 mins	3-4 Hrs
Disk Space	10 Mb	30Mb
Min. Memory	640 Kb	1-2 Mb
Performance*	38.7 sec	100.3 sec
Price	£99.95	£800.00

*Byte Excel benchmark, 1000 iterations on a 20 Mhz 386. Hardware Requirements: 286, 386 or 486 IBM compatible PC with a hard disc & a high density floppy drive. The current release of Coherent does not run on PS1, PS2 or other Microchannel machines, or systems with non-Adaptek 154X series SCSI controllers.

"Mark Williams Co. seems to have mastered the art of illusion; Coherent comes so fully qualified as a UNIX clone, you find yourself thinking 'I can't believe it's not UNIX.'"
UNIX Today! 26 November, 1990.

"... (Coherent) may be the best thing that has happened to UNIX yet."
PC Week, 5 November, 1990.

"I recommend it (Coherent) for anyone wanting to get a feel for UNIX without that sinking feeling in the wallet." Personal Computer World, March 1991.



If your wondering how something as good as Coherent could come from nowhere. Well it didn't. It came from Mark Williams Company, people who've developed C compilers for DEC, Intel, Wang and thousands of professional programmers.

It's Amazing Value!

Yes, we know that £99.95 may be hard to believe. But it's true and we've also made it easy for you to find out for yourself. With our 30-day money-back no-quibble guarantee.

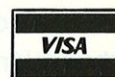
Our No-Quibble Money Back Guarantee

You have to be more than just a little curious about Coherent by now. So why not just do it? Pick up that phone and order NOW. You'll be on your way to having everything you ever wanted in UNIX, and for a lot less than you ever expected.

Call our order lines

Phone: 021 766 8333

Fax: 021 766 5968



AET Ltd, Suite 2, Bradford Court
131, Bradford Street,
Birmingham B12 0NS

*Plus shipping and VAT. Coherent is a trademark of Mark Williams Company. UNIX is trademark of AT&T. Xenix is a trademark of Microsoft.

work. This is sufficient for orthogonal projections but not for perspectives. Figure 5 illustrates what can happen. If you are looking down on a tall building from close to, the top will appear bigger than its base, even though the sides of the building are

parallel. In these circumstances, a line traced out along the ground up one edge of the building and along the top can assume a Z- or S-shape and so become double valued. Similarly, a function with steep cliff edges (eg the Mandelbrot mountain) even

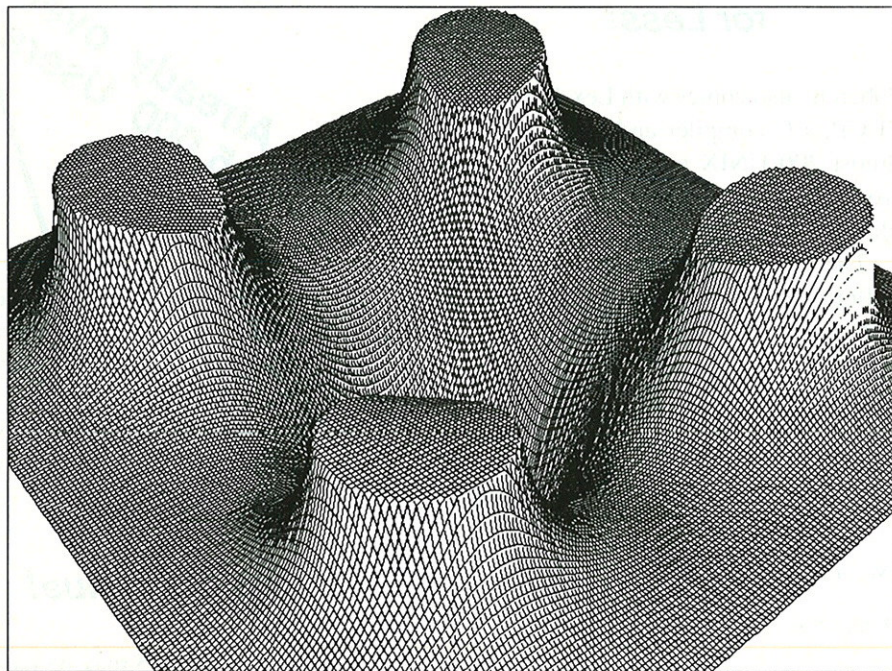
though itself single-valued can give rise to apparent overhangs when viewed in perspective from some view points.

The sample functions: the fun thing to do is to experiment! The three functions offered are just for starters. Before compiling your program rename one of them as Func. Suggested ranges of x - and y -values and values of viewing parameters are given in Figure 6. Broaden your horizons!

EXE

Dr Graeme Webster was formerly Head of Department of Computer Science and later Deputy Director, Academic, of Teesside Polytechnic. He has been involved with computer graphics for the last 20 years with an especial interest in 3D visualisation for Designers. He is currently setting up a Centre for Scientific Visualisation under the aegis of the Teesside Development Corporation.

The code given with this article, together with the SuperVGA library and Dr Webster's other graphics routines, are available on disk. Send a blank floppy disk to the Editor, following exactly the instructions given on Page 1, column 1. If you do not follow the instructions, your disk will join the Editor's collection. Mark your envelopes '3D-GRAPHICS'.



The four spikes function

Ridgeway Press

13 Horseshoe Park
Pangbourne
Berkshire RG8 7JW

COMPUTER MANUAL

PRINTING

Specialist printers of

**Technical Publications,
Manuals, Instruction Books**

Ring binders/ slip cases plain or printed

Wire-o-binding, tabbing

Shrink wrapping, disk labels

New manuals or reprints. All sizes

Data conversion

Full artwork/origination service

RAPID TELEQUOTE SERVICE

**RIDGEWAY PRESS, 0734 845331
FAX 0734 845186**

CIRCLE NO. 994

MULTIPLE INTELLIGENT COM PORTS FOR OS/2

THE IMPAC 8 & 16 INTELLIGENT COMMS CARDS

- * MICROSOFT COM X COMPATIBLE
- * 8 TO 64 FULL MODEM PORTS
- * REDUCE CPU OVERHEAD FROM 40% TO 1% PER PORT
- * FITS ALL ISA (PC/XT/AT) MACHINES
- * DOES NOT USE SHARED RAM
- * PROVEN FIELD PERFORMANCE
- * UK DESIGN AND MANUFACTURE
- * DRIVERS FOR DOS, OS/2, UNIX & XENIX

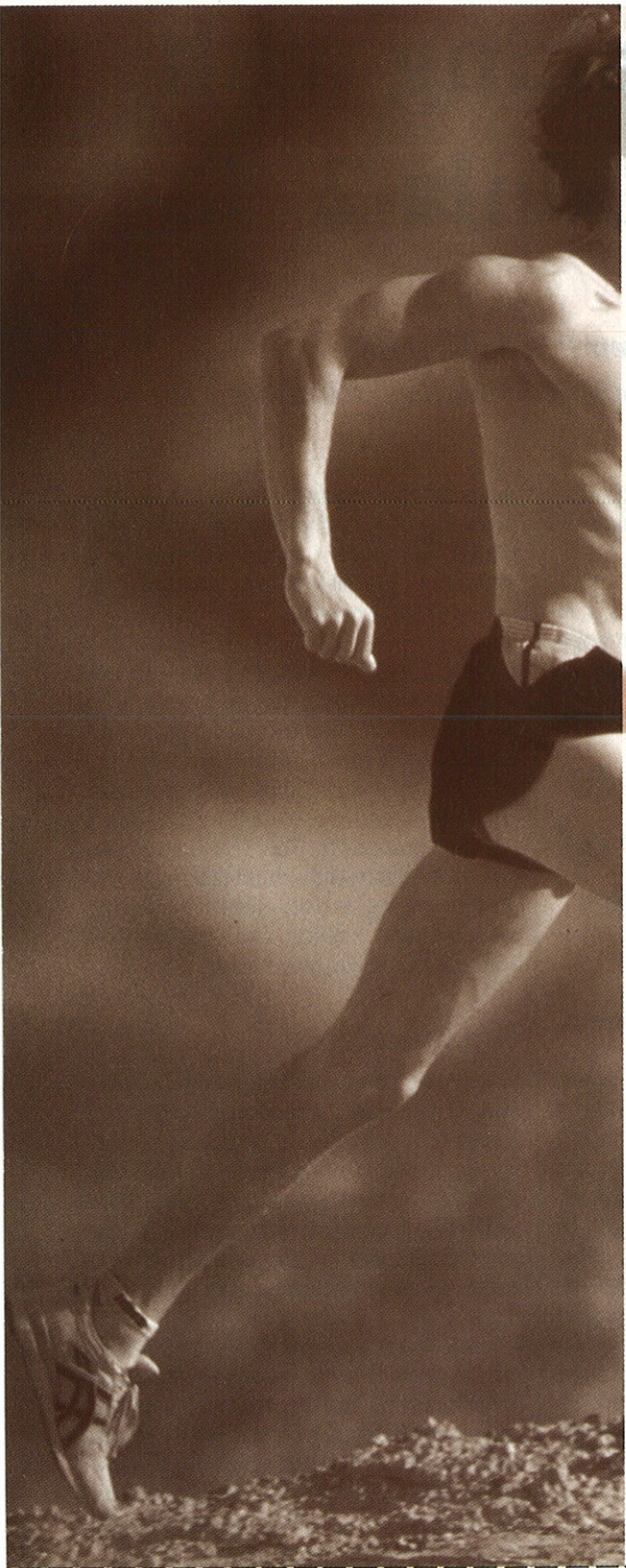
DEALER ENQUIRIES WELCOME

**For more information contact:
SCL Downwood, Carlton Road
Reigate, Surrey RH2 0JQ
Telephone: 0737-762200**

CDOS DRIVER ANNOUNCED

CIRCLE NO. 995

Runs on everything... well, almost



Glockenspiel C++ Compilers for UNIX

Glockenspiel C++ runs on the widest range of machines, from 386 PC's to the largest VAX. With over 120 supported machine/operating system configurations, including DEC, RS6000, Suns, SCO and many others, chances are we have the one you want. All Glockenspiel C++ Compilers are AT&T C++ conformant and all Glockenspiel C++ code is consistent across platforms.

Free C++ for your PC

With Glockenspiel C++ you can win the software development race; and because every race should have a prize we will give you a free copy of Glockenspiel C++ for DOS or OS/2 with every copy of Glockenspiel C++ for UNIX you buy (offer lasts until 31 August 1991).

For full details call The Products Group on
0285 - 655888

glockenspiel C++

QA

Please send more information on:

☐ Glockenspiel C++ for UNIX
☐ C++ for PC's

☐ CommonView Class Libraries for Motif, Windows or PM
☐ Training ☐ Consultancy

CIRCLE NO. 996

NAME

TITLE

COMPANY

ADDRESS

TEL. NO.

.EXE 7/91

Mail to: The Products Group, QA Training Ltd., Cecily Hill Castle, Cirencester, Gloucestershire GL7 2EF. UK.
Tel: 0285 655888 Fax: 0285 640181

MS-DOS Multi-tasking with TopSpeed Modula-2

Don't keep your users waiting! Make their PCs do 50 things at once with Richard Pickard's version of the TopSpeed Modula-2 scheduler.

Now that micros have become as powerful as yester-year's mainframes, we have the same opportunities to use our microprocessors to handle more than one application at once. SideKick and the Norton tools let you interrupt the current job, do a bit of work and then resume where you left off. These are examples of a simple kind of multi-tasking: *you* are the scheduler.

In more advanced systems, different programs with work to do may be carrying on even when the user has switched away from them. DESQview, UNIX, OS/2 and Windows are the best-known providers of this facility. What is going on here is multi-programming or 'heavyweight' multi-tasking (see Figure 1 for a glossary of terms) but now it's the operating system, or some addition to it, which is doing the scheduling.

There is a mixture of objectives in establishing this kind of scheme: it's partly to give the user the ability to move easily from one incomplete job to another and partly to allow free exploitation of the hardware by programs which are ready to do so. (If we switch our heads around, we could call the user 'a piece of hardware' and say that programs may be waiting for that resource at different times.) Under these arrangements, jobs which are interdependent must use operating system services explicitly to compete for or share resources.

With some systems, and Windows again is an example, each application can be organised so that parts within it can work concurrently. When the parts of a single application are being executed concurrently the multi-tasking is called 'lightweight'. The parts of the application will be mutually dependent, or at least related. They will compete for and share resources such as memory space, screen windows, the

keyboard and the disk; using their own arrangements and with minimal help from the operating system.

TopSpeed Modula-2 V2.X provides a facility by which you can organise an application program as a set of concurrently

What I did was to define an array of queues. Slick or what?

executing tasks. The facilities are provided by the Process module; it includes a time-slicing scheduler. Within your application, the sections of code to be executed asynchronously are written as parameterless procedures. Process module facilities allow you to set up and take down tasks (with chosen priorities); to wait for, release and test semaphores (to protect critical sequences); and to delay a task for a specified period. It's up to you to make sure that your tasks communicate and collaborate so that resources are correctly used.

Coroutines

The TopSpeed implementation is based on the use of coroutines. Properly speaking, a coroutine facility only provides for each task to schedule another task. The scheduling of tasks is organised directly by the tasks themselves; the task management system looks after the housekeeping of registers and stacks. This so-called 'hand-off' sche-

duling is efficient and easy to follow. The problems of resource-sharing can be simplified too, since - except for hardware interrupts - no task is ever unknowingly pre-empted by another.

The TopSpeed Modula-2 SYSTEM module contains a coroutine facility provided by IOTRANSFER (to establish hardware interrupt handlers), NEWPROCESS (to establish a new procedure as a potential coroutine) and TRANSFER (to switch among coroutines). The Process module adds lightweight multi-tasking based on time-slicing; it does this by exploiting the coroutine method and combining it with use of the hardware timer.

Organisation

Most conventional applications start by initialising variables, acquiring resources (files etc) continue by processing a sequence of commands, records or other signals, and finish by releasing the resources and stopping. With multi-tasked applications, the pattern is the same except that, in any of these phases, activities may progress concurrently.

In some database applications, a main record has detail records linked to it; the detail records may themselves act as links among main records. If you have had a job like this then you'll know that deletions can be tricky and - if done on-line - obtrusive. You can organise your database so that flagging the main record as 'dead' removes it and its details from view; why not then send a message about it to a garbage collection task which can get rid of it and any of its redundant details without too much interference to the on-line task?

Have you never had a client who asked for the statement total to be put at the top of

OPEN SYSTEMS PROGRAMMING & PORTABLE SOFTWARE

Use Pecan's Power Software development system and gain these substantial benefits.

- *Open systems product*
- *Portability of applications + information*
- *Protection of investment in software*
- *Choice of compiler USCD Pascal, Modula-2, Fortran 77, 'C'*
- *Substantial Cost Savings*

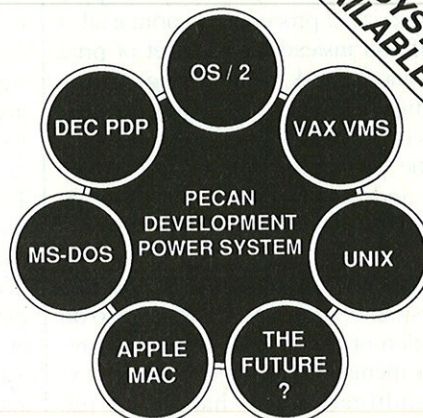
TURBO PASCAL USERS!! Open the doors to new markets

Move your MS-DOS restricted Turbo Pascal programs to Unix quickly and easily with our Pascal Liberator product.

- *Helps translate Turbo Pascal code to UCSD Pascal code which will then run under UNIX, MS-DOS, CPM, OS/2, APPLE the choice is yours.*
- *Pascal Liberator operates via a user friendly interface, offering pop-up help screens and split screen windowing to display differences between Turbo Pascal and UCSD Pascal command structures.*

Liberate your software for only £230.00+VAT
(This price includes a Power System)

32 BIT VLM SYSTEM
NOW AVAILABLE



THE ONE COMMON FACTOR

Get ahead of your competition
CALL US NOW

PECAN™

Telephone 0272 - 425012
Pecan Software Europe Ltd. Victoria House,
10 Kellaway Avenue, Bristol BS6 7XR. England.
Fax 0272 - 245000.

or
Distributor Grey Matter Telephone 0364-53499

CIRCLE NO. 037

MINIX 386 UPGRADE

Runs 386 Machines in Protected Mode
NO Code Size or Memory Restrictions!
Supplied Complete With GNU GCC Compiler & Manuals

For Only £125

Standard MINIX 1.5 available for :-
ATARI, IBM PC/XT/AT/386, AMIGA & MACINTOSH
RUN A UNIX TYPE SYSTEM ON ANY PC

For Only £125

DEMO COPY £15

FULL RANGE OF BOOKS ON MINIX/UNIX AVAILABLE

REGISTERED USER SUPPORT SYSTEM

Including printouts & patches from USENET.
Quarterly newsletters and telephone support.

Programs to run under MINIX: TEX, Metafont, Emacs, Rolodex,
Bison, Yacc, Spell & Look, Spreadsheet, RCS, Range of GNU Software.

M-KERNEL update for ATARI & AMIGA MINIX £48.25p

OPERATING SYSTEMS book 719 pages (The Minix Bible) £23.95p

HARD DISK DRIVES for IBM's 20 to 256 Megabyte
50 Mb £230-80 Mb £395-125 Mb £475-256 Mb £635-1.2 Giga Byte £1850
All disks carry a 12 month warranty

COMPUTERS

BUILD YOUR OWN 386. full range of 386 boards from £300
or buy assembled in **DESKTOP** or **TOWER** case.

IBM 386sx clones available from £990 33Mz dx from £1500
VGA Cards, Monitors, Memory Upgrade SILs and CHIPS available

ACCESS - VISA by PHONE, SAME DAY DESPATCH
P&P £4.80p

S.A.E. or phone for full details, ACADEMIC PURCHASE ORDERS
WELCOME. All prices plus v.a.t.

THE MINIX CENTRE
FORNCETT-END Nr NORWICH NORFOLK
ENGLAND NR16 1HT
TEL/FAX :- 0953 789345

MINIX is a trade mark of Prentice Hall. UNIX is a trade mark of AT&T

CIRCLE NO. 038



Anti-Virus

Utilities

*Probably the best
Anti-Virus package
you'll ever use!*

**BATES
ASSOCIATES**

**TOTAL
CONTROL**

contact:- Unit 3, Station Yard, Hungerford,
Berkshire RG17 0DY
Tel: 0488 685299 Fax: 0488 683288

CIRCLE NO. 997

the sheet? The simple solution is to build the whole statement in memory - as print lines - and call a procedure to print it all at once. If you instead pass the set of print lines to another task, then printing can proceed while the preparation of the next statement is in progress. (You could, of course, have the punter run the MS-DOS utility PRINT, and pass filenames to it - but we expect more *finesse* from .EXE readers.)

In data acquisition applications, interrupt routines can be overlong if they have to do application-oriented work rather than restricting themselves to device control. Exploit multi-tasking to handle all the application work and use data queues to pass messages to and from device drivers (see below).

In my simulation problems I have used quite large numbers of concurrent tasks to represent activities. Message queues hold details of jobs waiting to be processed. You could make one of the tasks a data-gatherer and another a visual realiser of the model.

Technicalities

Before you start your tasks, you must call `Process.StartScheduler` to ensure that the timer interrupt procedure is installed. (The initialisation section of the `Process` module installs the idler task.)

To do TopSpeed Modula-2 multi-tasking, you write the sections of code to be run concurrently as parameter-less procedures. After `StartScheduler`, use `Process.StartProcess` to initiate each procedure as a task and to allocate memory for each of their stacks; `StartProcess` adds a descriptor record for each task to a queue. This descriptor record incorporates the details necessary to establish the context needed to run the code. The queue is for 'ready' tasks and is maintained in priority sequence.

It is the combination of code and data which makes a task; a procedure may be used any number of times in different calls to `StartProcess`; `StartProcess`

will allocate the different data areas. Tasks may be added at any time. An existing task may remove itself (ie commit suicide) with `Process.StopProcess`.

Timesliced Scheduling

When you start the scheduler, a basic cycle starts in which, at each timer tick (approx 55 ms), the timer interrupt routine is entered as though it were a coroutine. The context for the interrupted task is saved in its data area on entry to the timer interrupt handler. The sequence then is:

The one-shot timer interrupt routine re-installs itself.

The routine places the descriptor for the current task at the end of the queue.

It removes the head of the queue to yield a descriptor for the next task to run.

Then it (re)initiates this next task as though it were a coroutine.

That is the basic cycle, but there are some wrinkles.

- The scheduler will prevent a new task from taking over if the interrupted task is in a critical sequence, ie it has called `Process.Lock` more often than `Process.Unlock`. Your procedures must call the `Lock` procedure in order to prevent concurrent use of system resources such as MS-DOS, memory management routines or global variables. Naturally, they must call `Unlock` afterwards to end these critical sequences. The `FIO` and `Window` modules - in the `MTHREAD` memory model - incorporate some use of this protection method, but you must use it explicitly around calls to I/O and similar DOS-based procedures, and around calls to procedures which rely on private data (think about `Lib.RANDOM`).
- Since tasks may have different priorities (lower numbers for higher priorities), the queue of ready tasks is organised so that 'end of queue' has several meanings, one for each priority.
- If a task is waiting for a signal or for a number of timer ticks then its descriptor is placed in a separate queue until the task is made ready by the arrival of the signal or passing of time.
- Some uses of `Process` procedures cause immediate scheduling of the next ready task, regardless of the timer.

concurrent	operating in the same period of time.
coroutine	a procedure which can interrupt itself by continuing the execution of another procedure and whose own execution can be continued by such other procedures.
critical sequence	a section of code whose execution must be protected from any interruption which might disturb the resources being used in the section.
grain	the amount of time between scheduling points spent executing a task. Crudely, coarse grain is measured in whole procedures and fine grain in machine instructions.
hand-off	a method of scheduling in which a task itself determines which is the next task to execute.
heavyweight	the scheduling used in multi-programming. This scheduling is heavy because the context switch is more involved.
lightweight	the scheduling used in multi-tasking. Light because context switches are cheap.
lock	in TopSpeed Modula-2, a data structure whose settings inhibit or permit different tasks to be scheduled. It is used to protect critical sequences. Generally: a data structure whose settings inhibit or permit different tasks to gain access to different resources.
multi-programming	a scheme whereby independent application programs are organised to execute asynchronously.
multi-tasking	a scheme whereby interdependent parts of an application program are organised to execute asynchronously.
process	the identifiable asynchronous execution of a procedure (it's identified by its stack and any allocated data).
ready	the state of a process that it could execute immediately if offered the opportunity.
round robin	an arrangement of sequencing the execution of processes that puts the newly ready at the tail of a queue without regard to priority.
schedule	to organise the sequence and durations of execution of a number of processes.
semaphore	a data structure used to organise communications among tasks. (It can also be used to protect critical sequences.)
simultaneous	operating at exactly the same time.
thread	an execution of processes in a sequence which deals with one packet of data. Also, loosely, a process.
task	a process.

Figure 1 - Glossary of terms

Compare Zortech v3.0.

DOS, Extended 16-bit DOS, Windows, OS/2, Extended 32-bit DOS, UNIX 386, and Macintosh.

land
nn Howlley
ect: Zortech C++ v3.0
Please find out more about
Zortech's new royalty-free
Sounds like a great de
Look into it and get ba

PRODUCT!
SAVE DTP
GROUP MONTHS
CROSS DEVELOPMENT
TIME!

RECOMPILE
PANDORA APP W/
ZORTECH'S
SCIENCE & ENGINEERING
EDITION -

FLOATING
POINT
→ IS ←
IEEE
748!

TELL CRAIG
ABOUT WINC
LIBRARIES!
URGENT

Data

9182

NOTES
↓

ZORTECH'S
ARE OPTIMIZED!

ZORTECH- MORE

* \$70 EXTRA
* LOT \$!!
* \$1000'S MORE

PLANNED -

* ZORTECH ON OS/2
V.2.0
(UP TO \$1295 EXTRA
W/ HARTH \$1000'S)

* UP TO 46 MEMORY

ZORTECH'S FAR
MORE EXTENSIVE!

PAY EXTRA \$'S
FOR BORLAND'S
LIMITED SOURCE

THE ROYALTY-FREE
DOS EXTENDER
COULD BRING
AVATAR PROJECT
BACK-ON-BUDGET

FEATURES -

C++ COMPILERS - DOS/WINDOWS
C++ DEBUGGERS - DOS/WINDOWS
WINDOWS SDK REPLACEMENT
WINDOWS SDK - HARDCOPY DOS.
WIND - DOS APPS → WINDOWS
16-BIT ROYALTY-FREE
DOS EXTENDER
OS/2 C++ COMPILER/DEBUGGER
32-BIT C++ COMPILER/DEBUGGER
32-BIT ROYALTY-FREE
DOS EXTENDER

C++ CLASSES

FULL LIBRARY SOURCE

ZORTECH BORLAND

Call 081-316-7777

For Your Information Pack

CIRCLE NO. 998

Delay (0) will de-schedule the current task. Tasks calling Unlock will be de-scheduled if there are no pending Locks.

Signalling

I've mentioned one rather limited example of task communication: the use of Lock and Unlock. It's a kind of broadcast command: 'Everybody freeze!'. The interesting communications are those in which tasks talk to each other and say things like: 'Another X is ready to be done', 'I've finished Z-ing', 'A key has just been struck' or 'I've found an error'.

The Process module provides a signalling system for this kind of communication; it is based on the idea of the semaphore. A Process.SIGNAL is implemented as a count of waiting signals and a priority queue for the waiting tasks' descriptors. The procedures are Init (to set up a semaphore) and SEND, Notify and WAIT (to send and receive signals).

If a task waits for a signal before one has been sent, then the waiting task is placed in a queue, associated with the signal, until the signal arrives. If signals are sent before tasks wait then the counter is incremented. There are two procedures for sending signals: SEND and Notify. The difference is that SEND can cause the calling task to be de-scheduled if there was a waiting task and it had a higher priority. Process.Notify doesn't do this so it can be used from inside hardware interrupt routines.

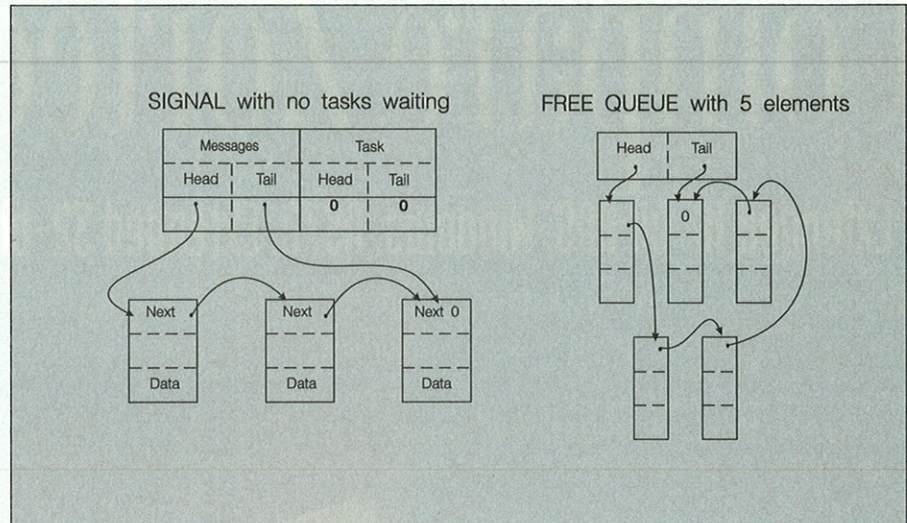


Figure 2a - Message and free queues

If queues of data are needed rather than indistinguishable signals, as in the classic academic, and in very real producer-consumer problems, then you might use a single semaphore to protect a global data structure - such as a mail box or list of queues - and build your own routines to store and access the data in the right sequence.

It is in the nature of tasks that they are cyclic; in a loop they wait for a signal then carry out some processing. If a task is waiting for an external event, in particular a keyboard event, then if the event does not occur the task should allow itself to be rescheduled immediately (eg: Process.Delay (0)). See Figure 3 for a sample of how this is done.

What more do we want?

The TopSpeed implementation is economical in that it provides a comprehensive service, within the limitations of DOS, and it does it with an easily-learned, small number of different procedures. Remembering that memory management, file handling and that DOS and BIOS calls are all provided for in other library modules, the Process module completes the set of essential tools for multi-tasking.

If it falls down, it is this very economy which trips it. The features are fixed, they cannot be parameterised or altered (eg: once started, a task's priority cannot be changed). In execu-

```

TYPE
  ElementPointer = POINTER TO QueueElement ;

  (* Queue elements are used both for      *)
  (* messages and as task control blocks. *)
  (* A queue is a singly-linked list.      *)
  QueueElement =
    RECORD
      QE_Next : ElementPointer ;
      QE_Cor  : PROCESS ;
      QE_Data : LONGWORD ;
    END ;

  (* Queue anchor. *)
  (* Head is NIL if empty queue. *)
  QueueBase =
    RECORD
      QB_Head : ElementPointer ;
      QB_Tail : ElementPointer ;
    END ;

  SIGNAL = POINTER TO SigRec ;

  (* Head-to-head queues of messages awaiting *)
  (* collection and tasks awaiting messages. *)
  SigRec =
    RECORD
      MsgQueue : QueueBase ;
      TaskQueue : QueueBase ;
    END ;

```

Figure 2b - Task descriptor and message TYPES

```

PROCEDURE QSwap (VAR FromQ, ToQ : QueueBase) ;
(*
  Swap the head element of FromQ to tail of ToQ,
  they may be the same.
  Must always called with interrupts disabled.
*)
VAR
  HoldIt : ElementPointer ;
BEGIN
  (* Note the element being moved *)
  HoldIt := FromQ.QB_Head ;
  (* Pop the head of From queue *)
  FromQ.QB_Head := FromQ.QB_Head^.QE_Next ;
  (* Push on tail of To-queue ... *)
  WITH ToQ DO
    IF QB_Head = NIL THEN
      (* ... q was empty *)
      QB_Head := HoldIt
    ELSE
      (* ... q already had a valid tail *)
      QB_Tail^.QE_Next := HoldIt
    END ;
  (* Note and mark end of to-queue *)
  QB_Tail := HoldIt ;
  QB_Tail^.QE_Next := NIL ;
END ;
END QSwap ;

```

Figure 2c - How queue elements are moved

KIBWORTH COMPUTER TRAINING

Language courses are normally held in groups of up to three and are tailored to suit individual needs:

- C** For structured and procedural programming (and inline assembler code).
- C++** A better C: Object-oriented programming and design
- Windows** Various ways of implementing event-driven GUI's, including Windows 3.

DATABASE Relational databases: PARADOX with PAL and the Paradox Engine

There is no need for add-on courses, and each self-contained course forms part of an overall design to train project-leaders needing a broad outlook.

Kibworth Computer Training
68 Springfield Crescent
Kibworth Beauchamp
Leicester LE8 0LH

0533 - 792653

CIRCLE NO. 034

EUROPEAN INFORMIX USERS GROUP

Informix is a trademark of
Informix Software Inc.

Forthcoming Seminars to be held
in Central London

12th Sept 1991 Multimedia systems
5th Dec 1991 Complex databases

For further details contact:

Bryce Campbell (0532) 738 244
Jeremy Russel (081) 446 6481
Jerry Nichols (071) 251 2128

CIRCLE NO. 035

Desktop FORTRAN

EXPERTS AGREE WHAT TO BUY – THEY JUST CAN'T AGREE WHY!

"The . . . FTN77 compiler has allowed us to port our main-frame applications to 386 micros with minimum effort. The speed of compilation is incredible! . . . The run time performance is fantastic, full support for 32-bit operations and the 80387 coprocessor means our programs run like lightning!"
Tony Fitzpatrick, Exploration Consultants Ltd.

"I find the FTN77/386 compiler to be fast and easy to use . . . We often download programs under development from a mainframe to the PC to run through this computer."

Rob McLaren
FTN77/386 User.

"Without the FTN77/386 compiler it would have been totally impractical for us to convert our million-line + power system analysis program (PSS/E) to run on a 386 PC."
Wayne B' Rells
Power Technologies Inc.

"FTN77/386 provides more intrinsic functions than any other: functions to manipulate strings, handle screen and keyboard I/O, read the date and time, parse the command line . . . Advanced file manipulation functions allow an FTN77/386 program to work with files in a C-like manner".
Scott Ladd,
Computer Language, Nov. 1988.

" . . . I am very impressed and have decided to shift all our production programs over to FTN77. . . ."

Dr. G.R. Chaplain
NCL Investments Ltd.

"The ease of use, speed, debugging facilities, dynamic libraries with resulting small .EXE files, and extensive library routines made it a joy to use."

H. Gordon Jensen
FTN77/386 User.

"By far the fastest and most complete FORTRAN compiler that I have ever seen or used in the DOS environment".
Tim Wool
FTN77/386 User.

"The debugger was very easy to use. There is an on-line help facility which made reference to the manual unnecessary."

. . . Mike Gunn and Arul Britto,
EXE Magazine, May 1989.

"For sheer speed and sleekness the (Salford) compiler stands out like a Ferrari among Yugos"
. . . Jack W. Crenshaw, Computer Language, May 1990

FAST, FRIENDLY & FULLY FEATURED - THE PERFECT MIX - CALL US NOW!

SALFORD SOFTWARE MARKETING LTD. MAXWELL BUILDING · THE CRESCENT · SALFORD M5 4WT.
TELEPHONE: 061 745 5678 · FAX: 061 745 5666
FTN77 386/486 – extending MS-DOS.

**UNIX VERSION
AVAILABLE NOW!**

CIRCLE NO. 999


```

PROCEDURE KeyTask ;
VAR
  Pressed ; BOOLEAN ;
BEGIN
  LOOP
    TASK.Lock ;
    Pressed := IO.KeyPressed ;
    TASK.Unlock ;
    IF Pressed THEN
      Deal with keystroke. *)
    ELSE
      Schedule some other task *)
    TASK.Delay (0) ;
  END ;
END ; (* Main endless loop)
END KeyTask ;

```

Figure 3 - Waiting for a keypress

tion, it will sometimes (unexpectedly) schedule another task when the current task makes a call to some Process procedures. Here are some other considerations which made me look at the possibility of tailoring the Process module.

The 'fair shares' problem

In heavyweight scheduling systems there are several trade-offs and rules to deal with them. Different systems put a different emphasis on the balances between: response-time against throughput; CPU usage with I/O; operating system activity against application activity

and so on. In high-speed real time systems (as against low-speed, real-enough time systems) there are the additional trade-offs between acquiring all the data and processing what's already in; and in meeting the hard deadlines set by physical laws or by external, clock-driven devices.

One of the difficulties is that multi-tasking schedulers can't look ahead and tell which tasks are the most suitable to do next; they would be a lot better off if they could, especially in heavily loaded systems, because they could then prepare the 'best' list of imminent tasks so that all desires are most nearly satisfied.

I have often doubted the value of using priority as a basis for scheduling. In many applications, the tasks with high priority are very few in number and are quick-acting (or they can be arranged to be quick-acting). Hardware interrupts pre-empt the current task anyway. Low-priority tasks may be either quick or slow acting but, by definition, there is less time-pressure on them. For this situation I prescribe round-robin scheduling, with time-slicing but without priorities. And once a task has started its work, it hogs the CPU for the

whole of each slice or until it has finished its current job. The basis of this philosophy is that there is nothing better the system as a whole can do than what it is doing now because it is removing a load of work from the future. In this approach, the cost and number of context switches are reduced; the drawback is that the time to react to an incoming signal may be too long if there are lots of ready tasks and the 'high-priority' ones are some way down the queue.

To get round this possible drawback, the number of ready tasks must be reduced. The separate low-priority tasks - each originally designed to wait for and process one type of message - have to be combined into a single task which first decodes messages and then hands them on to the appropriate procedure.

Task-to-task messaging

The really interesting communications are those in which tasks talk to each other and say things like: 'Another X is ready to be done and here is its address', 'I've finished Z-ing and here's the buffer back', 'The "q" key has just been struck' or 'I've found error 0103H'. These are not mere signals, they are real messages.

```

DEFINITION MODULE TASK ; (* Multi-tasking procedures and data. *)
(* JPI PROCESS.MOD rewritten by Richard H Pickard. *)

(*# call (o_a_copy => on,
  o_a_size => on,
  near_call=>off) *)
(*# F_pdata *)
(*# call (seg_name => null) *)
(*# E *)
(*# module (implementation=>off) *)
(*# data (seg_name => null) *)

(* o All tasks have same priority - scheduling is *)
(* 'round robin' *)
(* o Signals are implemented by queues of LONGWORD messages *)
(* and queues of waiting tasks. *)
(* o Tasks are rescheduled only at calls to: *)
(* - Delay (including zero ticks) *)
(* - Wait (if no message is ready) *)
(* o Maximum delay is 1 minute (measured in ticks of *)
(* 1/16.2 secs) *)

TYPE SIGNAL ; (* Pointer to SigRec *)
(*# save *)
(*# data (volatile => on) *) (* Volatile vars, don't touch! *)
VAR LongestDelay : CARDINAL ;
(* due time of most future of delayed tasks *)
IdleCount : LONGCARD ;
(* Idler task - number of times invoked *)
SchedTime : LONGCARD ;
(* Number of ticks since start *)
(*# restore *)

PROCEDURE StartScheduler;
(*
  Start time-sliced scheduler; the calling procedure is one of the
  tasks. This procedure should be called before other processes are
  started.
*)

PROCEDURE StopScheduler;
(*
  Stop time-slicing. Tasks other than the caller will cease
  execution.
*)

PROCEDURE StartProcess (P: PROC; N: CARDINAL) ;
(*
  Create parameter-less procedure P as a new task, with N bytes of
  work-space, and add it to the tail of the round-robin ready queue.
  Caller is not rescheduled.
*)

PROCEDURE Delay (T: CARDINAL) ;
(*

```

```

  Wait T ticks; Calling task is always rescheduled, even if T = 0
  *)

PROCEDURE SignalSetup (VAR s : SIGNAL; HowMany : CARDINAL) ;
(*
  Initialise a signal s and create additional (HowMany) queue
  elements. The message queue is initialised as empty and the
  additional elements added to the tail of a system-wide free
  queue.
*)

PROCEDURE Send (s : SIGNAL; Message : LONGWORD) ;
(*
  Send 4-byte Message to first task waiting on signal s and add
  task to end of ready queue. If no tasks are waiting, put Message
  in FIFO queue (until some task waits for it). Sending task is
  not rescheduled.
*)

PROCEDURE Wait (s : SIGNAL) : LONGWORD ;
(*
  Wait for signal; if a message has already been sent, do not
  reschedule the calling task but return the message value
  immediately. If no message has been sent, suspend the caller in
  the signal's FIFO queue.
*)

PROCEDURE TaskWaiting (s : SIGNAL) : BOOLEAN;
PROCEDURE MsgWaiting (s : SIGNAL) : BOOLEAN;
(*
  Indicate if Task is waiting for message or Msg is waiting for
  task.
*)

(*# save *)
(*# call (reg_saved => (ax,bx,cx,dx,di,si,ds,es,st1,st2)) *)

PROCEDURE Lock;
(*
  Critical region lock - prevents time slicing so that DOS or BIOS
  calls can be made or global variables updated. Calls to Lock may
  be nested. This allows nested procedures to apply locks without
  regard to context.
*)

PROCEDURE Unlock ;
(*
  Unlock procedure (always paired with a call to Lock) ends
  time slice inhibition. Will not reschedule current task.
*)

(*# restore *)

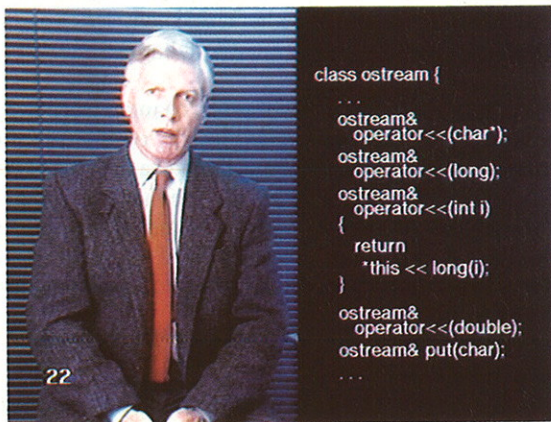
END TASK.

```

Figure 4 - TASK routines and data

VIDEO COURSE

Learn C++



NO RUSH!
NO TRAVEL!
NO HOTELS!
ONLY £299.95

Learn C++ Now!

The great exodus of programmers from C to C++ has begun! Since C++ builds on C, it's the easiest OOP language to learn. That's why it's called "the language of the 90's".

Why the rush?

PRODUCTIVITY! Yes, C++ programmers can write programs in less time requiring less maintenance. Large projects become much easier to manage.

Unfortunately, learning C++ can be very costly. Classroom instruction is expensive even without the travel and hotel costs. Of course, not learning will cost you even more in the long run. Now there is an alternative!

The top C++ video tutorial at the lowest possible price.

The C++ video tutorial from Zortech is the ultimate C++ training tool for work or home at only £299.95. It comes on six VHS video tapes containing 32 clear, extensive tutorials.

Used in conjunction with the concise workbook and tutorial disk, you will find everything you need for fast-track C++ tuition.

The course is generic (i.e. compiler and hardware independent) and is available with or without the award winning Zortech C++ Compiler for MS-DOS and OS/2.

Start writing C++ code within a week.

As a C programmer, you will start producing C++ code within a week of concentrated use of this course. Alternatively, spend just an hour a day watching the video and working through the suggested exercises to learn C++ in only six weeks!

The Leader in video tutorials.

If you don't already know C, you can join tens of thousands of programmers world-wide who have learnt C with the Zortech Complete C Video Course, described as:

"An excellent bargain ...
... I heartily recommend"

Gary Ray, PC WEEK

Save your company thousands of £££'s.

One programmer can train for only £299.95, but you can train ten programmers for just:

C++ Course	£299.95
9 Extra Workbooks	£179.55
Total 10 students	<u>£479.50</u>

Yes! Only £47.95 each!
(With all the FREE refresher courses you need!)

Now, Zortech with its new C++ Video Tutorial has refined the art of video tuition and presentation even further for the 1990's.

IN EUROPE CALL:

44- 1-316-7777

Only £299.95 complete

- Six Videos with 32 lessons
- 256 page workbook
- Tutorial disk
- Compiler & hardware independent
- NTSC or PAL format
- Tax deductible

Don't delay, order now!

Just mail the coupon or call the order hotline for same day shipment.

USA: Zortech Inc.,
1165 Massachusetts Avenue,
ARLINGTON, MA02174
Voice: 617-646-6703
Fax: 617-643-7969

EUROPE: Zortech Ltd.,
106-108 Powis Street, LONDON
SE18 6LU
Voice: 44- 1-316-7777
Fax: 44- 1-316-4138

ORDER FORM

Please rush me these items:

Qty	Description	Price
—	C++ Video Tutorial	£299.95
—	Extra C++ Workbooks	£. 19.95
—	Zortech C++ Compiler	£129.95
—	Developer's Edition	£299.95

Please add £5 + VAT postage in UK
All UK orders please add 15% VAT
International shipping charged at cost

Name _____
Company _____
Address _____
Phone _____
MC, VISA or CHQ _____ Expiry _____
Card No. _____ EXE 7/91

In a variety of real applications, when tasks synchronise with each other, they want to send messages. These messages may be little bits of coded information (commands and responses) or blocks of data (expressed by some kind of handle). It is a most useful facility to be able to pass such messages among tasks without first having to build and test a mail box or other queuing arrangement.

An important philosophical point about this approach is that the waiting task does not have to make any tests or searches for a message once a signal has arrived ... the signal is the message, all is pre-ordained, no time is wasted. An important practical point is that it should cost no more to send a real message than the combined cost of sending a signal and finding any data associated with it.

In the business of merely waiting for time to pass - in order to fire the noonday gun or to make measurements on the system of which a task is part - the time scale and granularity are important factors. If time-scale divided by time-grain is high then a general purpose time queue is necessary (and TopSpeed Modula-2 provides it). If the ratio is small then a more efficient queue

arrangement can be constructed (specially if the grain is one tick).

My way

I made three major changes and one minor change to the `Process` module. They were originally implemented with V1.0 of TopSpeed Modula-2, and subsequently 'pragmatised' and recompiled under V2.0.

1. REMOVE PRIORITIES

First I copied the `Process DEFINITION` and `IMPLEMENTATION` modules to new ones called `TASK` and redeveloped this. I combed out all references to priority and removed them. For example: priority is a parameter in the add-a-task procedure `Process.StartProcess`, but not in its brother `TASK.StartProcess`. Where an `IF` or other condition relied on a comparison between priorities, I kept only the code which would be executed if they were the same.

I now realised that the idler task might be a nuisance. There had to be an idle loop of some sort - in case no real task is ready - but it is an 'ordinary' task. The original idler had

lowest priority; any other task would be run if ready. Now, with no priority to separate it, the idler might take up a whole tick doing nothing. As it was an ordinary application task there was no restriction on what it may do. My replacement calls `Delay (0)` in a loop, so that it was always and immediately rescheduled. If there were no other ready tasks then this was a suitable thing to do, it will run again straight away; if there were other tasks then - let's hope - the overhead of rescheduling the idler was slight enough.

These first changes were easy. The next two relied heavily on the use of queues. I was going to need a method of describing them and the elements in them; the elements were to be task descriptors and messages. The `TYPE` of a task descriptor is the same as a message, it is shown in Figure 2b.

2. SIGNALS WITH MESSAGES

I followed the `Process` approach in having a setup procedure to define a message queue. The first parameter is the message queue's handle, it's a pointer to a queue 'anchor'. The new, second parameter is used to specify how many messages of this type may be outstanding at any instant. An instance of

LOOK at all OPTIONS.

Software Protection - but which TYPE ?

- **EVERKEY II** — The Hard^{ware} Option?

SINGLE WIRE ZERO LOAD - Total Compatibility.
Not available in ANY other Key product.

- **EVERLOCK** — The Soft^{ware} Option?

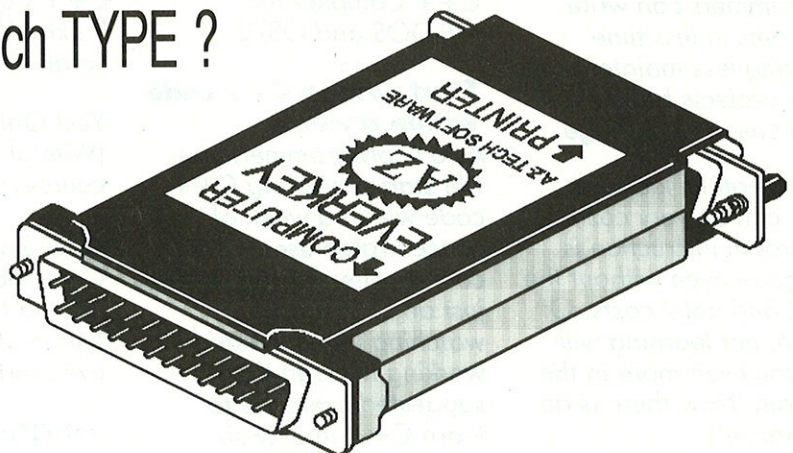
Software-Only Copy Protection; economical & SECURE.

- **EVERTRAK** — Non Copy-Protection Option?

Non Copy-Protection - Anti Hacker software protection.

CALL FOR A FREE WORKING DEMO DISKETTE

Other Services include:- Disk Duplication & supply - Label Production & supply - Packaging.



USER FRIENDLY
Micro Systems

22A, Bartleet Road,
Washford Industrial Estate,
Redditch, Worcestershire,
B98 0DG. England.
Tel: 0527 510 105
Fax: 0527 514 229

..... THE ONLY OPTION.

Sole Distributor for Az-Tech Software, inc. - UK, Ireland & Spain.

CIRCLE NO. 003

THE SOFTWARE CONSTRUCTION COMPANY

BORLAND

Borland C++	£199
Turbo C++ 2nd. Edition	£55
Turbo Debugger & Tools	£85
Turbo Pascal v.6.0	£69
Pascal Professional	£129
Pascal for Windows	£129
ObjectVision	£79
Paradox 3.5	£375
Paradox Engine	£215
Paradox SQL Link	£249

AUTHORISED

DATABASES

Advanced Revelation	£795
Advanced Revelation - askSam	£695
askSam LAN	£255
askSam LAN	£579
Clarion Personal Dev. 2.0	£79
Clarion Professional Dev.	£679
DataEase	£485
DataEase - LAN 3-PAK	£485
Dataflex	£449
Dataflex Multiuser	£1065
Knowledgeman/2	£645
Magic PC	£325
Open Access III	£449
Open Access III Compiler	£385
R:BASE 3.1	£515
R:BASE Compiler	£190
R:BASE 3.0 LAN Pack	£645

DBASE

Clipper 5.0	£299
dBASE III PLUS	£395
dBASE IV	£495
dBASE IV Developer's Ed.	£695
dBFAST/DOS	£255
dBFAST/PLUS	£255
dBXL	£165
dBXL LAN	£389
Force	£449
FoxBASE+	£295
FoxBASE+/386	£425
FoxPro	£515
Vulcan	£320

OBJECT-ORIENTATED

Smalltalk/V	£65
Smalltalk/V 286	£129
Smalltalk/V Windows	£325

CASE TOOLS

EasyCASE Plus	£190
EasyCASE Plus Prof.	£255
EasyCASE Prof. Upgrade	£80
Visible Analyst	£385
Visible Analyst Workbench	£2460
Visible Analyst Workbench	£6995
Visible Analyst Workbench	£1295
Visible Dictionary	£595
Visible Prototyper	£595
Visible Rules	£595

VERSION CONTROL

MKS RCS 5.1	£165
MKS RCS 5.1 - 5 User	£645
PVCS Professional	£319
PVCS - 5 User	£1345
SMS	£395
Sourcerer's Apprentice	£175
Sourcerer's Apprentice	£429

TEXT AND UNIX TOOLS

MKS AWK	£65
---------	-----

MKS AWK DOS & OS/2	£129
MKS Programmer's Platform	£429
MKS Toolkit	£165
PolyAWK	£129
PolyAWK Toolkit	£145
SpellCode	£65

TRANSLATORS

BASTOC	£515
BAS_C Commercial	£579
dBx/DBPort	£389
dBx/DBPort with source	£645
FOR_C	£359
Pascal to C Translator	£99

Microsoft

BASIC PDS 7.1	£219
Visual BASIC	£99
C 6.0 Compiler	£210
C 6.0 & Windows SDK	£349
COBOL 4.0	£345
FORTRAN 5.0	£199
MASM 6.0	£65
Pascal Compiler	£149
QuickBASIC	£50
QuickC	£50
QuickC/Assembler	£85
QuickPascal	£50
Windows 3.0	£65
Windows S.D.K.	£260

Authorised

EDITORS

BRIEF for DOS & OS/2	£199
BRIEF upgrade	£60
BRIEF with dBRIEF	£275
BTags	£35
Cheetah	£159
dBRIEF for dBASE/Clipper	£99
dBRIEF for Paradox	£99
dBRIEF for R:BASE	£99
EDT+	£190
EMACS	£210
Epsilon	£129
KEDIT	£99
MKS Vi	£99
Multi Edit Professional	£115
Norton Editor	£50
SPF/PC	£159
SYNDIE	£319
VEDIT PLUS	£120

LINKERS

.RTLink	£190
.RTLink/Plus	£319
ALINK	£129
Blinker	£125
Plink w/LTO	£359

MAKE UTILITIES

MKS Make	£99
MKS Make - 5 User	£320
PolyMake	£129
PolyMake - 5 User	£485

PARSERS AND LINTS

MKS Lex & Yacc	£165
PC-Lint	£90

PROFILERS

.RTLink/Plus	£319
--------------	------

CHARGE	£65
Codesifter	£79
Inside!	£85
PC Metric	£129

DEMO BUILDERS

Dan Bricklin's Demo II	£190
Instant Replay III	£99
Instant Replay Prof.	£385
Show Partner F/X	£259

MEMORY MANAGERS

386 MAX Version 5	£85
Above Disk	£79
Blue Max	£100
HEADROOM	£85
HEADROOM Network Extens	£39
HI386 Complete	£65
MOVE'EM	£59
QEMM-386	£65
QEMM-50/60	£65
QRAM	£55
Turbo EMS	£65

DOCUMENTATORS

4c for Brief	£79
ABC Flowchart	£190
ASMFLOW	£129
C-Clearly	£85
C-Doc	£120
Clear for C	£129
Clear for dBASE	£129
dAnalyst for C and C++	£190
dAnalyst for Clipper	£190
dAnalyst for dBASE III	£190
dAnalyst for dBASE IV	£190
dAnalyst for FoxBASE +	£190

FREE Microsoft Flight Simulator

WORTH £45 WITH
EVERY PRE-PAID
ORDER OVER £100*

EasyFlow	£99
Flow Charting III	£165
SourceDoc	£190
The Documentor	£190
Tree Diagrammer	£65

DISASSEMBLERS

Sourcer 486	£85
Sourcer w/BIOS pre-proc.	£110

E-MAIL

cc:Mail	£450
cc:Mail EXPAND	£385
cc:Mail Remote	£190
DaVinci eMAIL	£645
DaVinci eMAIL Combo	£965

CODE GENERATORS

DATABOSS	£395
Matrix Layout	£195
ObjectVision	£259
ObjectVision for Windows	£259
PRO-C	£259
R&R Code Generator	£99

UI2 Developer's Release	£385
UI2 Touch & Go	£259
Zachary	£259

DEBUGGERS

Periscope 286/386 Pod	£389
Periscope I OK	£319
Periscope I/512K	£385
Periscope I/MC	£449
Periscope I/MC 512K	£449
Periscope II	£145
Periscope II-X	£129
Periscope IIw/Switch	£145
Periscope IV (16 MHz)	£775
Periscope IV (25 MHz)	£900
Sherlock	£129
SVS DBG	£210

MULTI-TASKING

DESQview 2.2	£85
DESQview 386	£125
Dr. Switch-OnCall	£65

BATCH ENHANCERS

Batcom	£39
Beyond.BAT	£65
Command Plus	£85
ProKey Plus	£85
The Builder	£99

TERMINAL EMULATORS

BLAST II	£195
COTERM/100	£145
COTERM/220	£160
COTERM/220	£160
COTERM/4010	£165
EM 320	£199
EM 4010	£230
EM 4105	£330
EM 4105 Network	£345
ICE.10	£190
SmarTerm 125	£190
SmarTerm 2392	£65
SmarTerm 240	£225
SmarTerm 320	£129
SmarTerm 340	£225
SmarTerm 400	£115
SmarTerm 4014	£145
Smarterm 470	£190
ZSTEM 100	£65
ZSTEM 220	£99

BRIEF 3.1

- Mouse Support
- EMS Support
- Dual mode DOS & OS/2
- Dual media 3.5" & 5.25"

**FREE Logitech
Serial Mouse with
BRIEF 3.1 still
only £199***

*All prices exclude V.A.T. U.K. carriage included. **One offer at time of purchase, per order at published prices.** VISA or ACCESS accepted.

Credit to approved accounts.
SUMMER '91 CATALOGUE
Call for your free copy of our 160 page catalogue, TOOLBOX.

**1 THE MALTINGS, GREEN DRIFT, ROYSTON, HERTS. SG8 5DB
TELEPHONE (0763) 244114 FACSIMILE (0763) 244025**

Windows3.0 ToolBox

Power up your Windows apps!

Over 200 functions!

With over 200 functions, Windows Toolbox is designed to enhance, simplify and dramatically speed up the development of Microsoft Windows™ applications.

Flatten the learning curve.

The growing popularity of graphical user interfaces and the release of Windows 3.0 means more and more programmers will be entering the Windows environment, but the learning curve associated with Windows development is steep. Windows ToolBox short circuits this development cycle, providing the higher level functionality that Windows applications require.

Seven new classes!

Time and *date* classes are accompanied by a variety of alternative display styles (12 and 24 hour time formats and the various international date strings). Related functions convert time and date functions to and from strings.

For costing and financial applications, the *money* style adds refined capabilities over and above those of float data types.

Alistbox class is supported with numerous functions to ease user selections and lock out costly data entry errors.

The *edit* class makes manipulation of text a breeze, while the *integer* class expands upon the primitives supplied in the Microsoft SDK.

To display processing, ToolBox adds a *gauge* class with thermometer style readouts.

£249.00

PCX
Effects & Text

IMAGE Solutions!

VBase Colour Database **£249.95**

Turn your dBase, Clipper, C, Basic, Pascal, application into a powerful 256 - COLOUR DATABASE with little programming skills. Includes full printer support! Most any application can be converted in just a few hours.

PCX Programmers Toolkit **£165.00**

Includes over 75 routines to display, save, scale, and print PCX bitmapped graphics from almost any programme. Full use of virtual buffers and EMS 4.0.

PCX Effects **£85.00**

Add spectacular special effects (F/X) to your program. Wipe, split, crush, slide, sand, drip, diagonal, spiral, random, or explode your graphics. Requires PCX Toolkit.

PCX Text **£149.00**

Display blazing bitmapped text in any graphics mode as simple as text mode. You can display your text information on top of graphics.

GX Graphics **£179.95**

A complete graphics library supporting all graphics primitives. Use the GX Graphics Toolkit instead of the BGI or MS libraries and make your program faster, smaller and more portable across compilers - while supporting more video modes.

**Pinna Electronics Ltd., APL Centre,
Stevenston, Ayrshire KA20 3LR**

Tel: (0294) 605296 or Fax: (0294) 68286

CIRCLE NO. 007

Zortech C++ 3.0

Zortech C++ for DOS & Windows 3.0 **£250.00**

Optimised Compiler Windows & Dos
Royalty free 16 DOS Extender
WINC (convert DOS command line
programs to Windows) Libraries

Zortech C++ Developers Edition 3.0

Compiles & debugs DOS,
WINDOWS, OS/2, DOS 386
Royalty-free 16 bit DOS Extender
Royalty-free 32 bit DOS Extender.
Library source included

C++ Tools **£400.00**

Zortech C++ Science & Engineering Edition 3.0

Support for DOS, Windows, OS/2 and
DOS 386

M++ Array Language extension with
Weitek Chip support

IEEE 748 Floating Point

Zortech Numerics Guide. **£600.00**

Free

**Zortech C++ Video Course
with every Zortech 3.0 order**

(R.R.P £299.00 plus VAT)

Offer expires 31 August 1991

Offer exclusive to System Science

Prices are exclusive of VAT, but include delivery

System Science

3-5 Cynthia St, London N1-9JF

Fax: (071) 837-6411

(071) 833-1022

CIRCLE NO. 008

4. LESS SWITCHING

My overall approach relies on the idea that tasks spend a fair amount of time waiting for work. When they get it the best thing for the system as a whole is to carry on with it. To promote this I finally removed all unnecessary reschedulings from the procedures in TASK.

This meant taking out the invocations of `Process.Slice`. This routine was freely used - in the V1.0 module - to schedule the next task even when no mandatory rescheduling point had been reached; I wanted to remove what I felt were unjustified context switches.

The advantage of plenty of switching is that keyboard-intensive work looks smoother (because the other tasks continually yield control). In my applications, the 'low priority' tasks use `Delay (0)` a lot.

The effect of this change, on keyboard-intensive applications, will be noticeable if there are several non-keyboard tasks each of which does long-lasting bits of work; 'long-lasting' means more than a tick or two. Our confidence in a supposedly 'instant response' device drops if its reaction

time goes much above 100 ms or it becomes irregular; we would certainly notice keyboard delays of four or five timer ticks. Careless use of either of TopSpeed's Pro-

We would certainly notice keyboard delays of four or five timer ticks

cess or my TASK module could induce these effects; and, with either, they can be avoided.

As long as you've got your head round the basics of how coroutines work, and what the original `Slice` routine does then you can make your own version of the `Process` module. If you think you need 8-byte messages (or none at all); you want delay times in whole minutes; you want to keep priorities; or you fancy putting pro-

cessor-load measurement into the idler task; these are the modifications you can now make yourself.

EXE

Richard H Pickard spent a significant part of his early career doing operating system, database management system and programming language development and retains a keen interest in the techniques of computing alongside his main activity of applying computers to business problems. He can be contacted on 0525 61836.

TopSpeed Modula-2, for DOS and OS/2, and with a range of toolkits, is distributed by Jensen & Partners; call 0234 267500 for pricing details.

The author wishes to thank Jensen & Partners for [correcting his errors in the original article and] permission to quote from the Modula-2 library source.

We do not have enough space to print all Richard's code, but we can copy it onto disk for you. Please send us a blank, formatted disk plus return mailer, following the instructions given on page 1. Mark your envelopes 'MODULA-2'.



UK: 0223 462131
Int'l: +44 223 462131

Training

OSF/MOTIF™ and X™


► Learn how to build graphical user interfaces for UNIX software from the experts



OSF/Motif Programmer's Workshop
This popular 4 day workshop concentrates on using the Motif toolkit to build applications with OSF/Motif look and feel. Includes full course materials and extensive hands-on.



X Programmer's Workshop
Xlib, the base layer for all UNIX GUIs, is covered in detail during this 4 day hands-on workshop. Complimentary to Motif course for those writing their own widgets.



X Strategic Overview
An intensive one-day seminar on X looking at marketing, commercial and technical issues. Ideal for sales and marketing personnel or those overseeing company strategy.



On-site Training
For larger project teams, IXI provides these courses on-site. Courses can be tailored to meet your particular needs. Call Kim Fawkes at IXI Sales for more information.




IXI Limited 62-74 Burleigh Street Cambridge CB1 10J England Tel: +44 223 462131 FAX: +44 223 462132

* OSF/Motif is a trademark of the Open Software Foundation, X is the trademark of MIT.

T R A I N I N G

DOS - UNIX "QICPAK - ONE"

1/4" DATA CARTRIDGE INTERCHANGE DOS-UNIX/-DOS

- Time/Date stamping of files preserved
- On the fly end of line conversion
- Full mapping of DOS/UNIX file names with optional creation of unique names under DOS
- Handles device and linked name entries
- Compatible with Apollo, Mentor Graphics, Sun, SCO UNIX etc.
- Full programmers interface.
- Source code of VTAR, Vutil, Vtrans supplied in Microsoft C.
- 200 page reference manual, controller card and ribbon cable.
- Software only and complete solutions available.
- Optional facilities to process and duplicate cartridge from other systems.
- Support is by the developers.

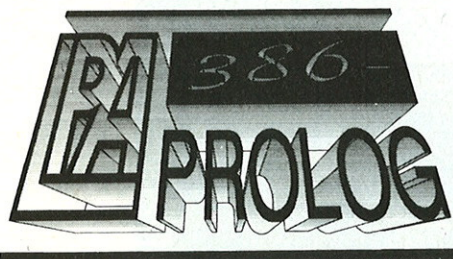


VOGON
ENTERPRISES LIMITED

94 Easthampstead Road,
Wokingham, Berkshire RG11 2JD
TEL: 44 (0)734 784511/890042
FAX: 44 (0)734 890040

CIRCLE NO. 010

Think BIG!



Forget 640K. Forget EMS. Forget the 16-bit squeeze. With LPA 386-PROLOG, you get a genuine 32-bit programming environment, which runs directly under MS-DOS and yet is capable of addressing up to 4G (4096M) of memory. All you need is a 386 or 486, and the talent to Think BIG!



Logic Programming Associates Ltd
Studio 4, Royal Victoria Patriotic Building
Trinity Road, London, SW18 3SX, England
Tel: 081 871 2016 - Fax: 081 874 0449

CIRCLE NO. 011

F77L-EM/32

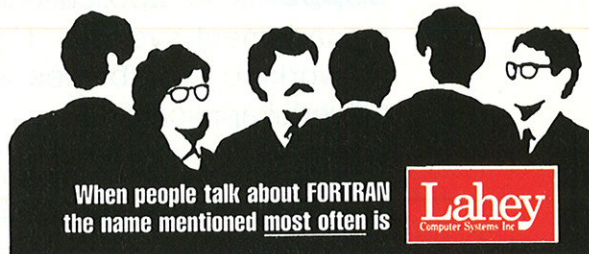
Port 4GB mainframe applications to 80386s with this 32-bit DOS-Extender compiler. The winner of PC Magazines's 1988 Technical Excellence Award just got better. New Version 3.0 and OS include: Editor, Make Utility, Virtual Memory Support, DESQview Support, New Documentation and Free Unlimited Runtime Licenses. F77L-EM/32 and OS/386. POA.

F77L

The compiler of choice among reviewers and professionals. Includes a Debugger, Editor, Profiler, Linker, Make Utility, Weitek and 386 Real-Mode Support, Graphics. POA.

Lahey Personal Fortran 77

New Version 3.0: Full ANSI 77, Debugger, Editor, Linker, Library Manager, Microsoft and Borland C interfaces, 400 page Manual. POA.



When people talk about FORTRAN
the name mentioned most often is

Lahey
Computer Systems Inc.

3-5 Cynthia Street,
London N1 9JF
Tel: (071) 833 1022
Fax: 081 837 6411

System Science

CIRCLE NO. 012

RM /Cobol-85

RRP £1050.00

- Ver 5 adds features incl. pop-up windows
- GSA certified ANSI Cobol-85 compiler
- Compiles to portable object code.
- Available for DOS, OS/2, AIX, Unix, Xenix.

RM /Panels

RRP £330.00

- Forms design and Screen Management system
- WYSIWYG Rapid screen prototyping

RM /CO*

RRP £170.00

- Application development environment
- Edit, compile debug and project management.

Special Offer

RM Cobol-85, RM/Panels and RM/CO* for
DOS at our very special price of £995.00 plus
VAT, only from System Science

Prices are exclusive of VAT, but include UK delivery

System Science

3-5 Cynthia St, London N1-9JF

Tel: (071) 833-1022 Fax: (071) 837-6411

CIRCLE NO. 013

dBASE USERS!

**How many days would it take to program, test and debug this system?
With Sycero and Clipper you could do it in HOURS!**

Clipper is the applications development standard for PC database management systems. Clipper's open architecture and superior networking capabilities let you create more sophisticated and powerful applications.

Sycero is a complete applications generator for Clipper. Making use of all of Clipper's advanced facilities, Sycero lets you develop PC applications in a fraction of the time it would take to do manually.

– Pulldown, bouncebar and Novell style menus produced in minutes.

Overlay a browse window with a function key in just five lines of code.

Automated transaction handling.

Memo fields give you word processing in a window on your screens.

Complete & powerful report generator for forms, lists and more complex invoices/statements.

Sycero with Clipper – the perfect development environment.



CIRCLE NO. 014

Please send me further details about Sycero & Clipper:

Name:

Address:

.....

Tel: Fax:

System C Ltd., 60-61 High Street, Maidstone, Kent ME14 1SR
Tel: 0622 691616 Fax: 0622 691241

.EXE 7/91



A Tough Apprenticeship

Jules May is worried about where the next generation of computer programmers will come from.

While poking around in my loft the other day, I found my first computer. It was pretty pathetic by today's standards, but it was not bad for its time. I built it myself, because in those days that was the only way to get a computer one could play with.

It had 128-bytes of RAM. It wasn't a lot, but it was enough to do useful things in. Back then, computing was regarded as a branch of electronics, and it was taken for granted that one worked close to the metal. Indeed, there was not even a word to distinguish between this kind of work and the abstractions we use today. In common with all the programmers I knew then, I knew the entire instruction set by heart, programmed in hex, and could read paper tape. Thinking back, I can't even remember what I did with it, so divorced was it from today's machines.

As I left school, computers were just being put in. Schools were standardising on BASIC, not because it was a productive way to work, but because it was easy to teach. As a result, a whole generation of teenagers graduated from school thinking they could program. Off-the-shelf applications were very thin on the ground, and were expensive - games even more so - so few pupils got the chance to see what commercial software looked like, (and few teachers as well, if truth be told). A lot of small businessmen had nasty experiences with computers that had been programmed for them by these whizz-kids.

Putting computers into schools was not a bad idea, although it could have been done much better. Once someone understood what a program was, and how it did its work, the colleges and universities had a much easier job teaching him to write real programs, on real machines. Some institutions even rose to this challenge and did teach these things! But the real learning came on the job - a wannabe would get employment with a company which would immerse him in the correct environment, and he would pick up concepts like defensive programming, complexity, and idle-time

processing from people who were conversant with such ideas (even though, back in those days, such ideas still had not been named). The gap between the programmers and the machine was not that great - perhaps one layer of compilation, and a library or two - so the trainee could easily get the feel of how his code was working.

How do you program something like 'F16' in BASIC?

I feel very lucky, because I became aware of computers while they were still very young, and comparatively little real work had been done. I have been well placed to follow the developments in computer technology, because I have had time to understand one development before the next one came along. For someone coming into the field now, the story is very different - computers are now so sophisticated it is barely possible to follow this track. Pupils at school see arcade games in cafes, they see computer games at home, and there is no way to connect that kind of thing with what is being taught to them. How can you program something like 'F16' in BASIC? Modern software is fast, pretty and largely mode-less, and none of the established teaching methods can cope with that, being (as they are) almost entirely goal-oriented.

So, how do you teach people to write programs? Our wannabes have decided they want to be programmers, and they have been taught BASIC instead, so, they go to college to learn the next step. The problem now is that the colleges don't know what the next step is either - the field is moving so fast that by the time something has been crystallised into a course, it is already out of

date. All the stuff that used to be learned on the job is there, and that is certainly useful, but the students still are no nearer to being able to write commercial software.

The next step (for those people who *still* want to program for a living) is to get a job. From everyone's point of view, this is the hardest step of all. Now, more than ever, giving a job to a graduate is buying a pig in a poke - the employer can have very little idea whether the applicant is going to be any good at the job, because the applicant is unlikely to have anything remotely relevant to show him. From the applicant's point of view, to be part of a programming team, no matter how small a part, there is so much he has to know. The computers are immensely complicated (and few companies program for just one machine now), there are layers and layers of insulation between the programmer and the machine, and huge libraries which have to be understood before any work at all can start. The only way to learn all this is to do the job, but you can't do the job until you know it. How does anyone learn to program these days?

I do know a few people who have completed (or have nearly completed) a formal education, and are now very good programmers. In every case, though, they are good not because of anything they have been taught, but because they have an instinctive feel for what is going on, and their education has filled in gaps rather than provided the foundation. As time goes on, and computers get more complex still, there will be fewer and fewer such people.

EXE

Jules May, among other things, teaches programming and computer animation to art students at a film school, few of whom are interested in logic or machinery. Because of the advances in computers, the job gets harder every term (for him and for his students).

Programmable Typesetting in TeX

Ever wanted to tell your DTP exactly where it gets off? TeX reaches the parts other systems have never even heard of. Peter Flynn is our guiding TeXie.

When desktop publishing first saw the light of day in 1978, computerisation was still a dirty word among many printers and typesetters. Although mechanisation of the process had been commonplace since the previous century, computerisation was still restricted to specialist equipment. The idea that you could replace the black art of composition by a program on an ordinary general-purpose computer was novel, to say the least.

And yet that is exactly what TeX did. Its inventor, Don Knuth, wrote it to ease the pain of having his famous 'Art of Computer Programming' re-typeset. Compositors are not known for their love of mathematical or technical setting, and charge accordingly. There were plenty of computerised typesetting systems around, but as they were all on dedicated equipment, specific to the manufacturers, they charged accordingly, too!

Far from being just a text formatter, TeX introduced the concept of programmable document structure as well as the idea of personal desktop publishing. The program was written in WEB, a structured documentation language which generates Pascal source accompanied by TeX-format documentation. TeX was thus immediately portable to a huge range of equipment, and versions in C and other languages (even FORTRAN) followed. It now runs on almost everything from Apple IIe/Amiga/Atari through PCs/Macs, UNIX workstations and VAXen up to IBM and other mainframes, even the Cray.

Commands are the keyword

TeX processes your text input file and outputs a binary typeset file, all justified, paginated and paragraphed, ready for a print

TeX runs on almost everything from the Apple IIe to the Cray

driver to go to work on. By inserting command keywords flagged with a backslash into the text, you can bring into play a limitless range of formatting capabilities. This in itself was not new: old die-hards will remember RUNOFF and its progeny, the original 'dot-line' word processors, but TeX's macro-driven programmability opened up a much wider choice of formatting capabilities, and because it works from a plain ASCII file, the source text is portable between all platforms.

The obvious and elementary facilities like paragraphing, justification, hyphenation and pagenumbers are built-in defaults (although you can disable them if you need to). TeX has, of course, the specialist mathematical capabilities for which it is famous,

but these can happily be ignored by most users. The real power comes from the facilities to handle the structure of a document: by using macros, an entire design can be controlled with a handful of mnemonic commands, and the details of the layout can even be decided by the program on the basis of the text you feed it, on the fly.

The advantage of this kind of programmability is twofold. First, you get a very high degree of reusable code: hence the existence of large style-file repositories. Second, you can make changes to the appearance of the document by changing the macro definitions: most of the time you need never touch the text itself, provided it has been adequately marked-up.

Mark-up

The distinction between TeX and most other DTP systems lies in this concept of mark-up. A fully marked text shows what the elements and structure are in generic terms: for example, a section heading is simply labelled as such, with the heading text clearly delimited: it is up to the style declarations to implement how it should appear in typographical terms. This approach dramatically reduces the time taken to achieve the accuracy and regularity expected of professional typesetting.

So-called 'visual' DTP (the familiar Page-Maker, Ventura and Quark) relies mainly on manual dexterity with a mouse and an eye for alignment, with some help from tag files (style sheets). The immediate visual feedback makes the idea of DIY typesetting attractive, as well as being superficially rewarding for the novice (and good for sales).

So-called 'logical' DTP (TeX, SGML, some word processors) use commands to label the elements of a document independently of their appearance, so the structure of the text is known to the program. You can view

```
\def\section#1#2{\flushevenpage\hbox{}\vskip1in
\centerline{{\bigroman#1}}

\vskip.3333in{\smaller#2\par}\vskip.5in\noindent}
\def\flushevenpage{\vfill\ejject
\ifodd\pageno\else\hbox{}\vfill\ejject\fi}
\def\smaller{\smallroman\baselineskip=9pt\noindent}
```

Figure 1 - Example definitions of macros

PROTECT YOUR PROGRAMS FROM PREDATORS

- Range of high security products
- High degree of transparency
- User friendly
- Support for IBM XT/AT/PS2 and Macintosh
- Parallel serial & bus keys
- Programmable version available with:
 - Four security levels
 - Wide range of language support
 - Software protection of .COM & .EXE files without need for source code.
- Competitive pricing (from £18.00)
- Free 30 day evaluation kit.

For further information and 60 day evaluation kit, contact:



COMSEC (U.K.) LTD.
Cranfield Airport,
Cranfield,
Bedfordshire,
MK43 0AL
Tel: (0234) 751203
FAX: (0234) 751082

COMSEC
(U.K.) LIMITED
COMPUTER &
COMMUNICATIONS
SECURITY SYSTEMS

CIRCLE NO. 015

What Prolog programmers have been waiting for !

Prolog-2 for Windows version 3.0

Once again Expert Systems Ltd. are first to deliver a Prolog system matched to today's programmer's needs.

Our new Prolog-2 for Windows 3 is the only package that allows programmers to combine the power of Prolog logic programming with the acclaimed GUI of Windows 3.

Now **you** can create 32-bit applications for the world's favourite GUI with **your** favourite programming language.

Just look what Prolog-2 for Windows 3 gives you :

- Full 32-bit addressing using 386 enhanced mode, no more limits on workspace, heap etc.
- Virtual memory support
- Startling speed. Up to 150 kLIP on a Research Machines VX486
- Full bidirectional C interface for embeddability
- Integrated window editor and graphics
- Over 500k of Hypertext help
- Workstation performance on a PC for just £1495.00

.....

I need a Prolog to write Windows 3 applications right now. Please rush me Prolog-2 for Windows 3.

Please add £10 p&p plus V.A.T.

My cheque is enclosed
or My official order is attached
or I wish to pay by
Mastercard/Visa
All trademarks acknowledged

copies at £1495 (£747.50 each)	
Card no.	
Expiry Date	
Name and address	
Signature	

Expert Systems Ltd.

Unit 12, 7 West Way, Oxford, OX2 0JB. England. Tel 0865 794474 fax 0865 250270

CIRCLE NO. 016

The MetroX X11.4 Server for UNIX System V 386 can now support 32,768 colours !

Upgrade to
16 Million Colours
soon!

Also Video Capture & Display
using X Windows

Graphics Cards supported:

EGA, VGA, SVGA, 8514, Targa 16, Targa+,
Hercules Graphics Station Card.

Runs on the following UNIX Operating Systems:

InterActive 386/ix, SCO UNIX, ESIX.

All Trademarks acknowledged

CIRCLE NO. 017

Powerful UNIX Systems for the serious developer £ 3,990

386-33 with cache
8 MB memory
120 MB, 18ms Hard Disk
1.2 MB Floppy Drive
2 serial, 1 parallel port
14 inch VGA Monitor
256 colours Graphics Card
Tower Case with 275 W power supply
UNIX System V
X Windows version 4

Call us now for your custom configuration

UniVision (UK) Ltd

PO Box 1098, Swindon, SN5 6LP

tel (0628) 822281 fax (0628) 822580


```
% TeX macros to check triangularity

\newcount\stdin \stdin=16 % unit 16 is the terminal
\newdimen\a \newdimen\b \newdimen\c % three numeric dimensions
\newdimen\twosidesum % counter to add two sides

\read\stdin to\firstside \a=\firstside % get the first side
\read\stdin to\secondside \b=\secondside % get the second side
\read\stdin to\thirdside \c=\thirdside % get the third side

\def\notatriangle{\message{This is not a triangle}\expandafter\end}
\def\equilateral{\message{This is an equilateral triangle}}
\def\isosceles{\message{This is an isosceles triangle}}
\def\scalene{\message{This is a scalene triangle}}

\def\triangletest#1#2#3{\twosidesum=#1 \advance\twosidesum by#2
    \ifdim\twosidesum<#3 \notatriangle\fi}

\triangletest\a\b\c \triangletest\b\c\a \triangletest\c\a\b

\ifdim\a=b
    \ifdim\b=c \equilateral \else\isosceles \fi
\else\ifdim\b=c\isosceles
    \else\ifdim\a=c\isosceles \else\scalene \fi
\fi

\fi

bye
```

Figure 2 - The Triangle Problem

the results on-screen after processing, but changes are done via the macro definitions, avoiding inaccurate manual placement.

Rolling your own

Defining a macro is straightforward. The principal construction is the `def` statement, which can have up to nine arguments, each being anything from single characters to whole paragraphs of text (or more). The definition follows in curly braces, containing the \TeX commands you want executed, and it is tokenised on processing and expanded on call, in the usual way of any macro. The easiest way to see how it works is to look at an example.

Figure 1 contains an example of some definitions for starting a new section of a document. Let us say there are two arguments, the section title and a text summary as a header, and we are printing two-sided, so we want sections to start on a right-hand page.

The call to `flushevenpage` causes the current page to be vertically filled (with white space) and the page ejected; if that leaves us on an even-numbered page, a further page-throw is made (floating white space at the top of a page is a rarity in continuous text, so it gets zapped in \TeX : a null `hbox` - horizontal box, here empty - is inserted to set a reference point).

We skip vertically one inch down the new page and centre the section title (the first argument) in whatever type-face has been loaded as `bigroman`. Then comes a third of an inch gap, and the summary set in whatever type setup is invoked by `small-`

ler. Finally there is a half-inch gap and an instruction not to indent the first paragraph.

There is a `font` command to name and load type-faces. The default, Computer Modern Roman (not Times, for once!) is based on Monotype 8a and answers excellently for general-purpose ease of reading continuous text, but \TeX will work happily with any PostScript, BitStream or METAFONT fonts.

TeX itself places great reliance on dimensional accuracy and precision (the same virtues dinned into compositors of old in a seven-year apprenticeship): you simply won't get misaligned indentation, wobbly margins or uneven spacing. If you ask for 12pt type on a 14.4pt baseline, then that's what you get.

And yet it is not rigid: dimensions like white space (paragraph gaps, inter-word spacing, margins etc) can be set to a value plus or minus a certain amount. This controls the

degree of flexibility of justification, ragged-right or ragged-left (unjustified) setting, and the automatic adjustment of paragraph spacing to fill pages to the right depth.

This kind of definition is not difficult to write, and it has the advantage of always giving a perfect fit, regardless of the text you feed it. The big plus, though, is that it obviates completely the need for fiddling with a mouse, and the dependence on screen-pixel resolution for positional accuracy (to say nothing of Monday morning handshake).

Triangles

To push \TeX a little further, the acid-test of the Triangle Problem is solved in Figure 2. All the usual programming features are used, with the exception of arrays, which do not exist, and formula translation, as in the definition of `triangletest`, where the `advance` command is needed to increment one variable by another. All `ifs` are matched by `fis`, and the use of macros to carry the output messages means the logic nest for triangle shape is kept very small and readable. One neat by-product of using \TeX is that measurements are denominated, so you give the sides of the triangle with their units, eg 3in, 55mm and 295pt.

The only sleight-of-hand needed is to make the routine exit cleanly if non-triangularity is detected, as this would occur within the domain of the `if` in `triangletest`. The `expandafter` causes `end` to be executed after the following instruction (the trailing `fi`) instead of immediately, thus terminating the routine without causing an 'unmatched-IF' condition (actually not a problem, but inelegant). This problem does not arise with the remaining tests, as they can all be made mutually exclusive in the one statement.

The interested reader may wish to examine one of Knuth's own exercises from the \TeX book (the bible for \TeX users), where he

```
\newdimen\x
\def\isoquote#1#2{\setbox1=\hbox{I}\x=#2
\setbox0=\vbox{\parshape=11 -0\x0\x -1\x2\x -2\x4\x -3\x6\x
-4\x8\x -5\x10\x -6\x12\x -7\x14\x -8\x16\x -9\x18\x
-10\x20\x
\ifdim\x>2em \rightskip=-\wd1
\else\frechspacing\rightskip=-\wd1 plus1pt minus1pt
\leftskip=0pt plus1pt minus1pt
\fi \parfillskip=0pt \tolerance=1000 \noindent #1}
\centerline{\hbox to\wd1{\box0\hss}}

\isoquote{I turn [...] his hand}{13.4pt}
\bye
```

Figure 3 - Macro for typesetting text in triangles

Xvale Ltd., Store 'D'
55, Bedford Court Mansions,
Bedford Avenue, London.
WC1B 3AD U.K.

TEL: 071-636-8210 / 071-631-4818
FAX: 071-255-1038 24 Hours

**Barclaycard
Access**

UNIX

*Now! you can get into UNIX and VPIX
(Multi-user DOS) at affordable prices, with a
leading specialist of the UK.*

Interactive Unix SysV, REL 3.2 Architect Series: (Ver 2.02-2.21)

SysV, Rel 4 CALL !

Workstation Developer: 1-2 usr £1295.00 U/L usr £1995.00

Operating System, Vpix (DOS under UNIX),
Software development system plus New 'C', CO-edit, Codewatch,
Ten-Plus, Looking glass, XII windows runtime and development system,
NFS/TCP/IP, and complete documentation.

Workstation Platform: 1-2 usr £855.00 U/L usr £1495.00

As workstation developer, excluding networking and development.

Network Developer: 1-2 usr £1155.00 U/L usr £1815.00

As workstation developer, excluding XII items.

Network Platform: 1-2 usr £695.00 U/L usr £1350.00

As Network Developer, excluding development items.

Application Developer: 1-2 usr £1095.00 U/L usr £1750.00

As Network Developer excluding network items.

Application Platform: 1-2 usr £495.00 U/L usr £1095.00

As Application Developer, excluding development items.

UNIX ITEMS

Motif Dev	£345.00
PC-DOS bridge	£195.00
Norton for unix	£195.00
Easy windows	£595.00
Ported Netware	£3995.00
SMB/ix	£895.00
Informix/Oracle	Call
Lotus 123	£595.00
LPI Cobol	£1095.00
LPI New 'C'	£595.00
LPI Fortran	£895.00
LPI Pascal	£895.00
LPI Basic	£595.00
LPI /PL/1	£1795.00
LPI RPGII Dev	£1795.00
LPI RPGII Util	£895.00
MENUIX	£49.95
SYSIX -pro	£145.00
LPI (unix) C++	CALL
XSIGHT	CALL

OTHER ITEMS

Xenix Items	Call
Unix o/sys	£525.00
Unix s/dev	£613.00
Unix vpix	£321.00
Unix tcp/ip	£227.00
Unix nfs	£340.00
Open dsktp	£795.00
O/dtp s/dev	£1095.00
O/dtp servrf	£1095.00
Guru reprinter	£49.95
Syseditor	£49.95
Cshell	£49.95
Menuix	£49.95
Sysix Pro	£145.00
SystemIX	£295.00
PC-Connect	£450.00
Pc-interface	£170.00
PC-TCP	£375.00
PC-Xview	£245.00
Terminl ctl	£695.00
Xvision	£295.00

LNX

Two hardware configurations + Software (a-e :FREE!.)

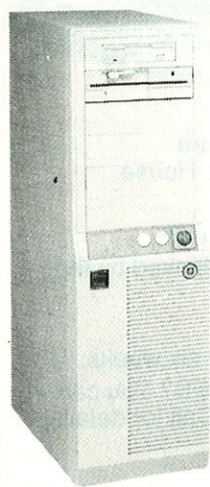
- 1) Entry level minimum mono hardware systems (a-b: free)*
- 2) Unix level higher spec hardware systems (c-e: free)*

Free Software Bundles

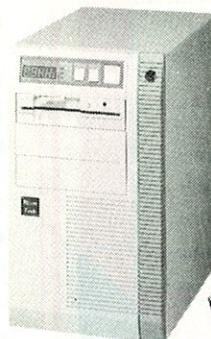
- a) Dos 4.01, Menuix, C-shell, Sysed under DOS, £225
- b) Unix, Sysix, Sysed, Dos, £475
- c)+b) Appl-Platform v2.02, £770
- d)+b) Net/Wrk-Plat v2.02, £1130
- e)+b) Wrkstn-Dev v2.02 £1570

HD/M'board	386-20	386-25	386-33	486-25	NB386-16	NB386-20
40 Mb	£895a / £1445b	£945a / £1695b	£1195a / £1845b	£1995b/£2790c	£1695b	£1895b
80 Mb	£1040b/£1690b	£1085b/£1810b	£1340b/£1995c	£2140b/£2935e	£1995b (60mb)	£2195c (60m)
120 Mb	£1110b/£1760c	£1225b/£1975c	£1410b/£2060d	£2210b/£3075e	** PRICE INFO ** Entry level followed by Unix level. (a-e) = free software	
200 Mb	£1185b/£1835d	£1300b/£2050d	£1475b/£2325e	£2275b/3200e		

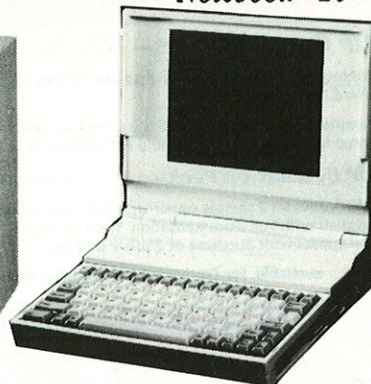
Unix level systems: Mini-tower, 1024*768 SVGA Colour screen, 102 keyboard, SVGA card w: 512k ram, 2Ser/1Par, Floppy/hard ctrlr, 5-1/4 AND 3-1/2 drives, 4MB RAM (486=8mb), Ami-Phoenix-Award BIOS, 12 mth RTB wty.



LNX systems



Notebook - 20



Entry level systems As for Unix except.: 1 floppy 1 Mb ram Mono screen & adapter

asks for the words of Pascal on triangles to be typeset into an isosceles triangle:

I turn, in the following treatises, to various uses of those triangles whose generator is unity. But I leave out many more than I include; it is extraordinary how fertile in properties this triangle is. Everyone can try his hand.

Knuth's solution is shown in Figure 3. The triangle is defined for up to 11 lines of type, which caters for values of x (which determines the angle of the apex) from 8.9pt to 47.2pt (triangles of between 11 and five lines tall), using Computer Modern Roman 10pt, 2pt leaded.

Comparisons

What amuses many TeX users is seeing the visual DTP systems busily re-inventing wheels that were rounded off many years ago. A recent review of PageMaker 4 for Windows (PCW, March 1991) lauded its introduction of controllable word-spacing and kerning, hyphenation, widow and orphan control, indexing and table of contents, as if these were something unusual,

whereas they have been around for 12 years! I see the page-limit is now 999: TeX's is 1,000,000. Embedded in this article is another on Personal Press which bemoans the difficulty of placing and adjusting graphics: mid-page and top-page floating inserts of your graphics are trivial in TeX, as indeed are automated forward and backward cross-references, bibliographies, multi-level footnotes, text colour separations, and a mass of facilities as yet unobtainable in most DTP systems. Many of these are handled by one of the most popular macro collections called L^AT_EX (for flexibility?).

One final test to whet your appetite for programmable control: typeset a perfectly rectangular paragraph (no indentation, and the last line filling out perfectly), actually quite difficult in most systems. In TeX, set `parindent=0pt` and `parfillskip=0pt` and away you go. Or set `parfillskip=\parindent` (with a non-zero `parindent`) and get diagonally symmetrical paragraphs (TeX justifies an entire paragraph at a time, not line by line, so you don't get unevenness or rivers - white space between words meandering down the page).

TeX does have one other advantage: it's free. Knuth placed the source code in the public domain, so there are PD implementations for almost every machine, with support available from user groups. If you want a more formal arrangement, there are also many excellent commercial implementations sold in the normal way. Details on all implementations are available from the UK TeX Users' Group, Computer Centre, Aston University, Birmingham (email uktex@aston.ac.uk) who can also supply addresses for other user groups around the world.

EXE

Peter Flynn is currently Manager of the Research and Academic Computing Service at University College Cork, Ireland. He has been Technical Consultant to a large City computer bureau, Deputy DP Manager for one of the UK Training Boards, and a Teacher of Programming and Systems Analysis. His hobbies are early music, reading and surfing. He can be reached by email as pflynn@bix and CIX, and through the wide-area networks as cbts8001@irucc-vax.ucc.ie

Zipit UP

PC SOFTWARE PROTECTION SYSTEM

- * Totally Protects MSDOS Programs from Piracy, Illicit Copying and Unauthorised Use
- * Easy to Use - No programming - Software only
- * Unbeaten Record of Applications to Products Throughout the Computer Industry Worldwide
- * U.K. Developed and Supported

ZipitUP Gives You All of This -

- * Very High Protection Levels
- * Totally User-Transparent and Friendly
- * Runs on All 3.5" and 5.25" LD and HD formats
- * No Expensive Add-ons
- * Use ZipitUP on your PC, PC Disk Autoloader or have your Duplicator Bulk Copy it for you

ZipitUP Core Utilities - £99.00 + vat

Includes Everything to Protect Programs onto ZipitUP Key Disks for Key Disk and Hard Disk Operation

PLUS 4 FREE Sample ZipitUP Key Disks and the Complete ZipitUP Reference Manual for all ZipitUP Utilities and Extra Modules

ZipitUP Evaluation Pack - £25 inc vat/UK pp

This Unique Offer provides 3 sample opportunities to protect your Software is supplied with Full Documentation. The cost will be Credited against subsequent purchase of ZipitUP utilities.

FREE ZipitUP Demo available. For Further Information and Brochure contact us.

THE **data** Experts in
BUSINESS Data Resources

The Data Business Ltd,
3 Bankside, Kidlington, Oxford OX5 1JE, UK
Telephone: (0865) 842224 Fax: (0865) 841507

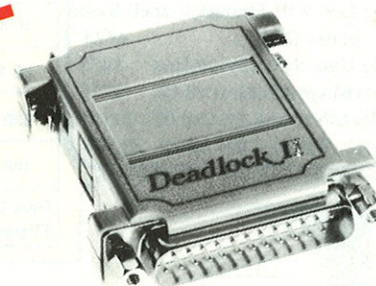
See Us On Stand 33 Software Tools '91

ZipitUP -
the Best Answer
There Is...

CIRCLE NO. 019

SPECIAL PROMOTION Deadlock-I

ONLY
£8.00
Limited Period

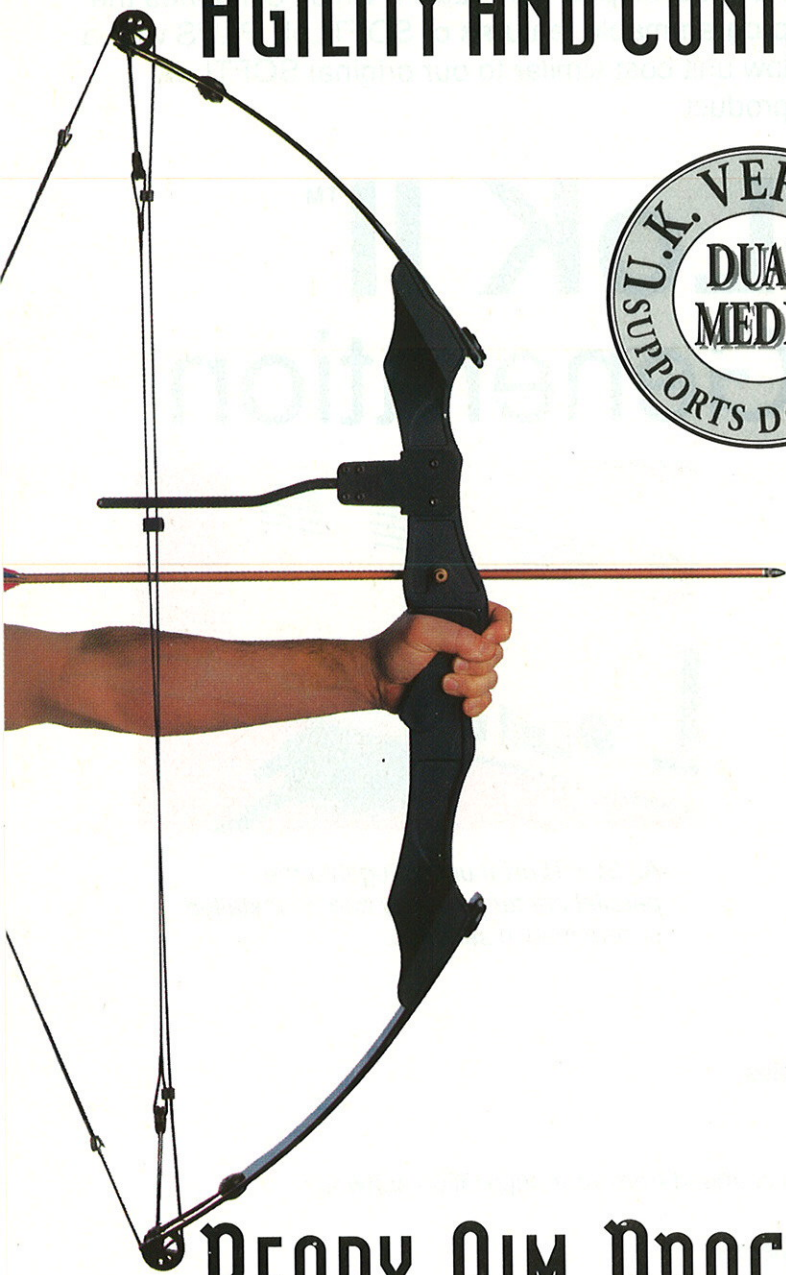



- Software protection units that are programmable by the Software House.
- No need to keep a stock.
- Ten (10) days delivery guaranteed.
- The codes can be changed if desired by the software house.
- Supplied with FREE software.
- Evaluation kit available free for six weeks.
- Are you already using DONGLES? You can have us as a second source. (Call for details).

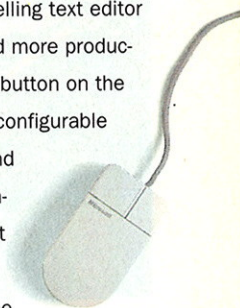
Please address all enquiries to:
B.L. Computer Security, 101 Hendon Lane,
Finchley, London N3 3SH, or call us on
081-343 0734 or Fax us on 081-346 2672




CIRCLE NO. 020

BRIEF'S NEW MOUSE SUPPORT GIVES YOU GREATER AGILITY AND CONTROL.




More than 100,000 programmers around the world swear by the virtues of BRIEF – its unmatched power, flexibility, and intuitive command structure. Now, thanks to mouse support and other new features, version 3.1 of the best-selling text editor will make you quicker and more productive than ever.  Hit a button on the mouse – a special, user-configurable menu window pops up and lets you execute any command you wish. Mark text and move around in your windows without using the keyboard. Zoom your windows to fill the

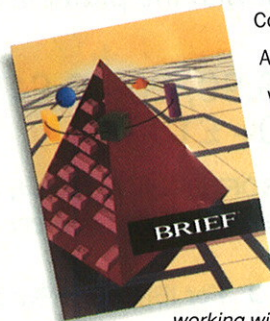


screen.  As snappy as it is, the mouse is only one enhancement among many. BRIEF 3.1 also takes advantage of EMS memory whenever it reads in a file. The result: faster performance, whether you're editing a file or running a macro.  So what else is new? Redo – a step for step complement to undo. Support for Microsoft C 6.0 Advisor and Microsoft's Programmer's Workbench. And more.  BRIEF's seamless interface with other Solution Systems products gives you instant access to the tools you need. A single keystroke lets you keep track of product development with Sourcerer's Apprentice – Version

Control for the Professional. Analyze product performance with CHARGE. And nothing beats dBRIEF for tailoring BRIEF to Dbase or Paradox.

 BRIEF sells for £199 excluding VAT.

 Once you start working with BRIEF 3.1 you'll really fly.



READY, AIM, PROGRAM.

Order your copy of BRIEF today.

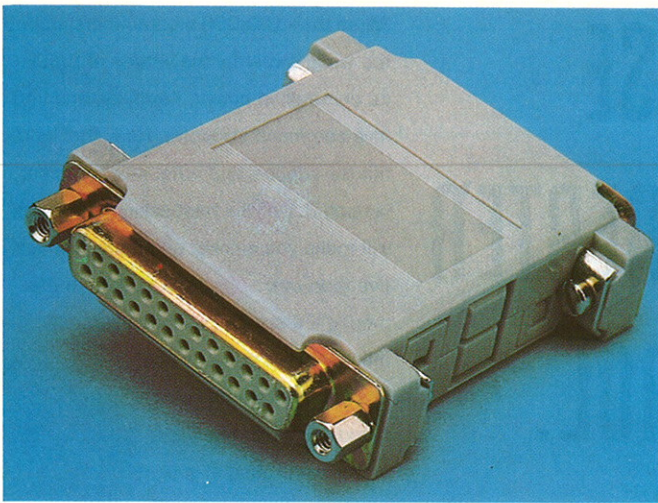
call:

0763 244141

©1991, Solution Systems. All rights reserved. BRIEF requires an IBM PC, AT, or 100% compatible, 256K RAM, and two disk drives.

CIRCLE NO. 021


Solutionsystems
THE PHYSICS OF PROGRAMMING
1 The Maltings, Green Drift,
Royston, Hertfordshire SG8 5DB
Telephone: 0763-244141
Facsimile: 0763-244025



How many users of your software paid for it???
SOFTLoK International Limited was established in 1987 with the introduction of our SOFTLoK and SOFTLoK PLUS devices to combat the ever increasing problem of software piracy. Our SOFTLoK range of software protection devices are used by hundreds of software developers from small consultancies to large multinationals. To cope with the ever changing needs of our customers we have developed SOFTLoK II which combines the programmable features of SOFTLoK PLUS with a low unit cost similar to our original SOFTLoK product.

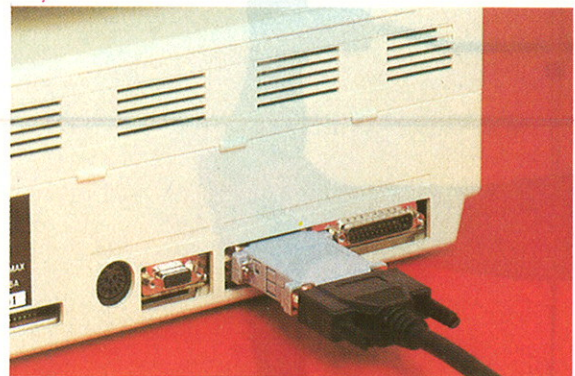
SOFTLoK II™

The Next Generation

SOFTLoK II units are programmable devices containing read/write memory protected by a password. Both the memory and the password can be changed at any time using our routines in your application software. Easy to use menu-driven software is provided to allow small or large batches of SOFTLoK II units to be programmed with their initial data & passwords ready to be sent out with the protected software product.

Price: 1-19 £16.90, 20-49 £15.40,
50+ £13.70

Evaluation kit £20 (SOFTLoK II, manual & software)
All prices excl. VAT and delivery



As SOFTLoK II units plug into the parallel printer port they can be installed or removed in seconds.

- ☐ For IBM PC, PS/2 and compatibles
- ☐ Uses parallel printer port
- ☐ Totally transparent to printer
- ☐ Secure data & password can be changed from your application software
- ☐ Cascadeable
- ☐ 240 bytes of secure read/write memory
- ☐ 8 byte (64 bit) password
- ☐ No programming adaptors required
- ☐ Easy to use SOFTLoK II setup software
- ☐ Routines ready to link with various compilers
- ☐ Easy to follow manual

I-MEX House, 40 Princess Street,
Manchester, M1 6DE, England,
Tel: 061 228 7379 Fax: 061 236 6890

Practical PostScript

PostScript printers are great, but it's really annoying to have to load up a heavyweight word processor, equipped with suitable driver, just to get a printout of your latest code. Barry Thomas has a work-around.

There are occasions when it's handy to be able to print out ASCII files on your PostScript printer without having to load up a DTP application or word processor in order

```
copy \post\before.ps com1:
copy %1 com1:
copy \post\after.ps com1:
```

Figure 1 - The POST.BAT batch file

to do so. Perhaps you just want to get a quick printout of a system configuration file such as AUTOEXEC.BAT, or you'd like to run out a draft copy of a text file which you are about to format. You could even use the technique described here to incorporate a simple PostScript driver into your own programs.

The utility assumes that you have your PostScript printer attached to the COM1 serial port of your PC, and that you have set the serial port MODE to the correct settings for

your printer, for example, using:
MODE com1:9600,n,8,1

This sets comms port 1 to run at 9600 baud, no parity, 8 data bits and 1 stop bit.

Installation and use

The suggested process of installing the POST utility is as follows:

Make a directory called POST.

```
/POINTSIZESIZE 12 def
/LINEHEIGHT POINTSIZE 1.15 mul def

% --- output buffers for 'show'ing
/BUFSIZE 5000 def
/buf BUFSIZE string def

% --- Margins for use in all pages printed
/LEFTMARGIN 80 def
/RIGHTMARGIN 530 def
/TOPMARGIN 750 def
/BOTTOMMARGIN 60 def

/EOLchar (\n) def
/SPACEchar ( ) def

% wordwrap? leaves true on stack
% if string exceeds right margin,
% false otherwise.
/wordwrap?
{ stringwidth pop currentpoint pop add
  RIGHTMARGIN gt
} bind def

% newline moves to new line
% showing a new page if necessary.
/newline
{
  currentpoint exch pop LINEHEIGHT sub
  dup
  % are we below bottom margin?
  BOTTOMMARGIN lt
  { % yes - new page
    showpage
    pop
    LEFTMARGIN TOPMARGIN moveto
  }
  { % no - move to new position
    LEFTMARGIN exch moveto
  } ifelse
} bind def

% printfile reads lines of text from
% the current input file up to
% the next EOF marker (\004) and
% prints it on the page, wrapping
% long lines and throwing a new
% page where necessary
/printfile
{
```

```
% loop, reading lines from current file.
{ % loop
```

```
% read line from current file
currentfile buf readline exch
dup
```

```
{ % loop: extract words from line
  SPACEchar search
  { % more words to come
    dup wordwrap?
    { newline } if
    % show word and space
    show show
  }
```

```
{ % last word in line
  dup wordwrap?
  { newline } if
  show newline exit
} ifelse
```

```
% word wrap rest of line?
dup wordwrap? not
{ show newline exit } if
```

```
} loop % words
```

```
% EOF found?
exch not
{ % pop off string and exit
  pop exit } if
```

```
} loop % lines
```

```
% show text on last page
showpage
```

```
} def

%%EndProlog
%%BeginSetup
```

```
% Change this font name if preferred
/Times-Roman findfont
POINTSIZESIZE scalefont setfont
```

```
% Move to top of page
LEFTMARGIN TOPMARGIN moveto
```

```
% Text to be printed follows the
% printfile procedure below
printfile
```

Figure 2 - The BEFORE.PS program

Copy BEFORE.PS and AFTER.PS into this directory.

Copy POST.BAT - given in Figure 1 - to wherever you keep your batch files, or some other suitable place on your PATH.

To print an ASCII file such as AUTOEXEC.BAT, you then just type POST AUTOEXEC.BAT

The batch file copies BEFORE.PS to the serial port of your PC. It follows that up with the given text file (in this case, AUTOEXEC.BAT), then finally copies AFTER.PS to the serial port.

Using the PostScript file BEFORE.PS given in Figure 2, the text file will be printed out in 12 point Times-Roman. Modify the source of BEFORE.PS if you prefer a different font or pointsize.

BEFORE.PS

This is the heart of the system. When installed in your PostScript printer, BEFORE.PS will make it behave like a simple line printer, printing newline terminated lines of text, wrapping long lines and throwing a new page if necessary.

Note that for input files containing lines longer than 5000 characters, this program will generate a rangecheck error when

If no EOF character is sent to the printer, the printfile procedure will wait for input forever

the readline procedure attempts to read a newline terminated line into buf. buf's maximum length is 5000 characters, controlled by the BUFSIZE declared at the top of the program.

The file AFTER.PS simply contains a PostScript end of file marker, CTRL-D (ASCII character 4). All jobs sent to your

```
^D
%%EOF
```

Figure 3 - The AFTER.PS file

PostScript printer *must* be terminated with the EOF character - in this case, if no EOF character is sent to the printer to terminate the job, the printfile procedure will sit waiting for input forever. You will have to check your word processor/editor manual to work out how to enter a CTRL-D character into an ASCII document. Well, you can't expect me to solve **all** your problems!

EXE

Barry Thomas works for Thomas Technical Publications Ltd (0335 70655) based in Ashbourne, Derbyshire, England. He is the author of A PostScript Cookbook, published by MacMillan.

This article was reproduced from the excellent PostScript Review (née PostScript Language Journal - International Edition) by kind permission of the publishers. The PostScript Review can be contacted on 071 792 9429, or by writing to 133 Notting Hill Gate, London W11 3LB.



DESkey THE COMPLETE DONGLE RANGE

Data Encryption Systems Limited have more experience in the design of software protection devices (dongles) than any other company in the U.K. 12 years experience in the design of dongles have gone into our latest 2 products: The DK1 and The DK12, both of which use the very latest in ASIC technology.

- 128 BYTES of memory split into 2 x 64. Half is intended for both read and write, the other half is read-only. Writing to this protected half can only be performed using your unique password. This feature can be used for tasks such as identifying the modules of a multi-module package which have been paid for.
- Down Counter. This enables a customer to be sold 'n' goes of the software. After which the DK1 stops working until the counter is reset using your password.
- Pseudo Random Number Generator. Billions and billions of bits without repeating! Software and Data Encryption could not be easier. 'Seedable' too.
- Completely Transparent to Printers, End Users and anything else requiring the port. 16 or more on any one parallel port.
- No Hidden Extras. Everything you need is in the DK1. No need for extra 'programming' units.
- DESlock Software uses DK1 or DK12 to provide instant encryption of .EXE or .COM software without the need for access to source (or .OBJ). Will even protect DOS commands. Keeps you/your employees the right side of the copyright laws.
- Designed and manufactured by the U.K.'s leading software security specialists Data Encryption Systems Limited.
- DK12 is a reduced feature DK1, available at a quantity price to beat ALL the competition (ask for a quote).
- Serial and Bus versions available.

For further information, contact us at,



Data Encryption Systems Ltd

Edbrook House, Cannington, Bridgwater, Somerset. TA5 2QE
Telephone (0278) 653456 Fax (0278) 653300

EIFFEL: The Key to Software Quality

The Eiffel System: A unique combination of advanced object-oriented techniques, for the quality-minded Software Engineer.

The Language: multiple inheritance, genericity, assertions, disciplined exception handling, polymorphism & dynamic binding, strong static typing, deferred classes...

The Environment: automatic compiler, source level debugger, automatic documentation tools, graphical design tools, garbage collection...

The Libraries: data structures, parsing & lexical analysis, persistency, X-Windows, character windows, interfacing tools...

Hundreds of companies worldwide have grasped the opportunity to produce quality reusable software, and are using Eiffel for their development projects - Philips, Telecom Australia, Motorola, Hawker Siddeley, GEC-Marconi Research... Join them.

For further information contact:
Applied Logic, 9 Princeton Court,
55 Felsham Road, London SW15 1AZ;
Tel 081 780 1088, Fax 081 780 1941

Applied Logic

CIRCLE NO. 024

THE TEXT FORMATTING COMPANY

TEX/LATEX Systems for workstations including Sun, Appollo Domain, HP Series 300, VAXStation, VMS and IBM PC's, Previewers, printer drivers and additional fonts are also available. C Source code can also be obtained.

NEW DVIPSONE - a printer driver for TeX which uses PostScript outline fonts. Produces PostScript files which can be sent to a typesetter/bureau for output. For further details contact:

The Text Formatting Company
Suffield Works
1 Suffield Road
London
N15 5JX
Tel 081 802 4470
Fax 081 800 5780

CIRCLE NO. 025

System Science

C COMPILERS

Microsoft C 6.0 with PWB	£245.00
Quick C & Quick Asm 2.51	£99.00
Quick C ver 2.5	£65.00
Borland C++	£239.00
Zortech C++ Dev Ed	£275.00
Watcom C 8.0	£255.00
Aztec C86 Comm	£265.00
C compilers for 386	£call

CROSS DEVELOPMENT

Introl C or Modula-2	£1445.00
Aztec C Cross	£275.00
IAR C Cross with CSpy	£call
2500AD Cross AsM	£135.00
Simulators & others	£call

FORTRAN

MS Fortran-77 5.0	£235.00
Lahey F77L-EM/32 & 386 OS	£845.00
Salford FTN77/386	£765.00
Lahey F77L	£375.00
RM Fortran Forte	£475.00

PASCAL, BASIC

Turbo Pascal (6.0)	£75.00
Object Professional (for TP)	£110.00
MS Quick Pascal	£60.00
Blaise Power Tools Plus	£89.00
Blaise Asynch Plus	£89.00
MS Basic 7.1 with PWB	£255.00
MS Quick Basic 4.5	£65.00
PowerBasic (ex Turbo Basic)	£85.00
MS Visual Basic (New)	£99.00
many Pascal, Basic libraries	£call

C DATA & MISC

Btrieve for DOS	£345.00
XQL for Btrieve	£475.00
Ctree Plus	£345.00
CodeBase 4.2 (source)	£175.00
CodeBase ++	£175.00
GL Financial Mathlib	£245.00
Blaise C Tools Plus	£89.00

C SCREENS & COMMS

Cscape, Look & Feel	£365.00
Vermont Views & Designer	£325.00
Panel Plus II	£275.00
C Worthy & cwARCHITECT	£call
Essential Comms (src)	£195.00
Blaise C Aynch Manager	£115.00
Greenleaf CommLib	£185.00

OTHER LANGUAGES

Smalltalk V/Windows	£295.00
TopSpeed Modula-2	£145.00
MS Cobol 4.0 (new ver)	£445.00
RM Cobol-85 ver 5.0	£995.00
LMI ur Forth	£225.00
PC Logo	£50.00
Clipper 5.0	£395.00

PROGRAMMERS TOOLS

SPE prog editor	£189.00
MKS Toolkit (Unix toolset)	£155.00
PVCS Prof	£375.00
PolyMake	£135.00
CProgrammer Toolbox	£159.00
Brief Editor 3.0	£195.00
Kedit Editor	£110.00
Flow Charting 3	£150.00
Dan Bricklin DEMO II (new)	£195.00
MS Progs Lib 1.2 CD	£225.00
Personal Rexx	£110.00

MEMORY, MATHS etc

386 Max 5.0	£75.00
QEMM/386 5.1	£60.00
GRAM for 286	£45.00
Above Disc (LIM Emul)	£69.00
Norton Utilities (new ver 5)	£110.00
Spin Rite II	£69.00
Derive (Symbol Alg) 2.0	£135.00
MicroStat II	£295.00
Mathematica by Wolfram	£call

WINDOWS 3.0

ASSEMBLERS & LINKERS

MS Macro ASM 5.1	£90.00
Turbo Debugger & ASM 2.0	£99.00
RTLink Plus	£345.00
Plink 86 Plus	£295.00
Sourcer & Bios (DisAsm)	£125.00
MS Windows 3.0	£85.00
MS Windows SDK & MS-C	£435.00
MS Windows SDK 3.0	£310.00
Case: W New Corp User	£CALL
WinPro/3 from Xian	£525.00
WindowsMaker	£595.00
Toolbook by Asymetrix	£295.00

"OUR RANGE INCLUDES MANY MORE ITEMS FOR
DOS, OS/2, UNIX ETC. PLEASE ENQUIRE"

3-5 Cynthia Street,
London N1 9LF
Fax: 071-837 6411

Prices are subject to change and alteration Prices include UK Delivery
but are exclusive of VAT

071-833 1022

CIRCLE NO. 026

Managing Data Structures

Hot-blooded young programmers like to code their own routines for everything. Peter Collinson reminds us of the riches of the UNIX library.

In the last few weeks, I have been sitting in front of my screen and generating C. It's a nice change to get back to real programming from time to time. I guess you never really *forget* how to write programs, they just get less clever as time goes on. You drop all those unreadable hacky things in favour of stolid dependable code.

Actually this happened to me some time ago. I *always* forget how I wrote something the instant I have finished. I have had to develop a coding style that allows me to make instantly intelligent defensive statements when someone thrusts some of my code under my nose claiming that there is something wrong with it. Of course, they are wrong. I quickly scan the stuff hoping that I will remember how it works, or how it was supposed to work, before the challenger realises that I have completely forgotten what I did, why, when, how etc etc. You cannot afford neat coding tricks with a leaky brain like mine.

Anyway, I was sitting with the code coming slowly off the ends of the fingers, and I realised how much use I make of the wealth of useful routines to be found lurking in the standard UNIX libraries. UNIX has always provided routines that implement useful algorithms in addition to those that interface directly to the system. Many of the algorithms are derived from *the* impeccable source: Knuth's three volumes 'The Art of Computer Programming'. However, the important aspect of this is that UNIX provides a *de facto* interface to some complex bits of code that you simply don't have to worry about. I thought that it would be interesting to talk about three of the routines that I use most often.

Sort

The sort routine has been around on UNIX since the earliest days. It performs an in-memory sort using the classical quicker-sort algorithm. The interface is simplicity itself:

```
qsort(bse, nel, width, cmp)
char *bse;
int nel;
```

```
int width;
int (*cmp)();
```

The first three parameters describe the data. It is lying in memory at an address contained in `bse`, it consists of `nel` elements of `width` bytes. The final parameter is a pointer to a user supplied function returning an integer. This function is called with two parameters that are pointers to individual elements in the data. It must return an integer less than zero if the first parameter is considered to be less than the second, equal to zero if the parameters are the same, and greater than zero if the first parameter is greater than the first.

So if we have a structure in a program like:

```
typedef struct Path
{
    char *name;
    char *path;
    int flags;
} Path;
```

where `name` is a text key onto some data, then to put the data into ASCII order we would write a comparison routine like:

```
int pcmp(a, b)
Path *a;
Path *b;
{
    return strcmp(a->name,
                  b->name);
}
```

If we have a base address to a vector of `Path` structures, we can sort them like:

```
Path *pbase;
int pct;
int (*pcmp)();

qsort(pbase, pct,
      sizeof (Path), pcmp);
```

If the number of elements is large, then the comparison routine will be called a great number of times and it's a good idea to optimise string comparisons. I often will check the first character of a string for equality before calling `strcmp`, writing something like:

```
int pcmp(a, b)
Path *a;
Path *b;
{
```

```
if (*a->name < *b->name)
    return -1;
if (*a->name > *b->name)
    return 1;
return strcmp(a->name,
              b->name);
}
```

There are some further optimisations that can be made with this too, but you get the drift I hope.

The `qsort` routine is now part of the ANSI C definition, so all good self-respecting C compilers should include this as part of their library. In some senses, then, it is no longer a UNIX routine. However, it was made part of the ANSI standard because it is embedded in many UNIX commands. As a result, it has come to be regarded as part of the programmer's standard armoury.

Hash tables

My `Path` data structure was taken from a real program that stores names of mail sites and the routes to them. One thing that I needed to do in the program was to store the names and search through them quickly to find whether a particular name had popped up before. The need to make data lookups based on text strings is an old requirement in computing. As assemblers and compilers began to have variables that were more than a single letter, then the compiler writer needed to find some method of looking up the details for the variable based on its name.

The idea is to take the name that has been used and perform some computation on its characters. The resultant number is used to index a table. If the entry at this point in the table is empty, then the name is unused. If the entry is full, then you have found the reference. It's not quite that simple, because different names will inevitably map onto the same point in the table and you have to cope with that in some way.

Tables managed like this are called *hash* tables and the technique is called *hashing*. This has nothing to do with herbs. Knuth



Security Software from Microft Technology.

M E N U G E N

MENUGEN provides a password protected menu interface. The system manager has complete control of what each user is permitted to do. Access to the DOS prompt can be prevented. MENUGEN has no memory overhead.

D - L O C K

D-LOCK prevents access to the hard disk after booting from diskette.

C L A M

CLAM provides TOTAL security. It incorporates MENUGEN and D-LOCK. In addition it provides a number of other facilities, the most important of which is file encryption – the only way to provide a high level of security on a micro.

Microft Technology Limited
The Old Powerhouse, Kew Gardens Station
Kew, Surrey TW9 3PS.
Telephone: 081-948 8255

CLAM £148+VAT Single User.
MENUGEN £48+VAT Single User.
D-LOCK £38+VAT Single User.
Corporate Licences available.

CIRCLE NO. 027

says that 'the verb "to hash" means to chop something up or make a mess of it; the idea of hashing is to chop up some aspects of the key and use this partial information as a basis for searching'. Knuth goes on to discuss the considerable body of known practice on hash tables: how to choose your hash algorithm to ensure that the function used to generate table indexes makes the data spread evenly throughout the table; and how to deal with the mapping of several keys to the same point in the table.

I have implemented this type of mechanism countless times. However, the System V UNIX world has had a standard routine set available to do this for ages; and if the routines are not on your system then they should be popping up as the move to System V, Release 4 looms.

The interface couldn't be simpler and comprises three routines. The table is created by a call to `hcreate` giving the maximum number of elements that you intend to store. The table can be destroyed by a call to `hdestroy`. The point here is that there can only be one hash table in existence at any one time. So `hdestroy` allows you to take down one table and start creating another.

The main access routine is called `hsearch`. This is passed a pointer to an `ENTRY` structure and a constant giving the action that is required. There is an immense repertoire of two actions: `ENTER` means that the data should be placed into the table at the appropriate point and `FIND` that no entry should be made but a pointer to an existing entry is returned if one exists.

The `ENTRY` structure is a pair of pointers:

```
typedef struct entry
{   char *key;
    char *data;
} ENTRY;
```

The data pointer can point to any data associated with the key; it doesn't have to be characters - these days the definition should probably be

```
void *data;
```

The header files were written before `void` was thought of. My program that uses these structures starts by calling `hcreate` and then has a loading routine like:

```
#include <search.h>
..
void load(pa)
    Path *pa;
{
    ENTRY e;

    e.key = pa->name;
    e.data = (char *)pa;

    (void) hsearch(e, ENTER);
}
```

The real program worries a bit about the return from `hsearch` being `NULL`, it means that the table is full. Notice how `e` is passed by value. The lookup code is:

```
Path *look(name)
    char *name;
{
    ENTRY e;
    ENTRY *r;

    e.key = name;

    if (r = hsearch(e, find))
        return (Path *) r->data;
    return (Path *)0;
}
```

The good thing about this is that I didn't have to worry about implementing the hard bit, the bit that does all the nasty string comparison and worrying about managing the hash table. I simply gave the code a key and a pointer. The routines did the rest. Annoyingly, I did have to worry about the maximum size of the table. This is due to the mathematics and the need to optimise the size to ensure that the data is distributed evenly. 'You canna bend the laws of Physics' as Scottie was fond of saying.

DBM

While `hsearch` is a routine from the System V world, the DBM routines have come down to us via the Berkeley derived systems. The original routine set was present on UNIX V7 and the author was Ken Thompson. But long ago someone in the UNIX System V camp decided to drop them from the distribution. DBM stands for Data Base Management which in truth is not really what these routines do. They allow you to store a key and data pair in a file (well actually a pair of files) and use hashing techniques to ensure fast data access to disk information.

An access function is used to map the key to a specific fixed sized portion of the file, often 1024 bytes. The effect is that given a key, the data can be found using at most two disk accesses. It is this speed that makes the DBM routines attractive to use especially if your key is non trivial, like a string or a complex data structure. For example, the *mmdf* mail system uses a DBM database to store its complex mail tables, knowing that the DBM routines provide fast keyed random access into the information.

The key does not have to be a text string, it can be the binary value of an integer or a data structure. Recent BSD systems have used the DBM routines to store the password data, and use the user login name and the binary uid as keys.

I use the routines to provide an index into an ASCII file for my personal name and address

list. I maintain the list using a standard UNIX editor, and then build a keyword index to the source file. Access programs use the index to seek directly in the source file, which speeds up performance considerably.

The DBM routines use the `datum` structure, defined in `<dbm.h>`

```
typedef struct
{   char *dptr;
    int dsize;
} datum;
```

Here again the `char *` is a hangover from older C. There is a pair of housekeeping routines: `dbmopen` that opens the data files and `dbmclose` that closes them. The `store` routine is given a key/content pair and writes the data out to the file. An approximation to my load routine above is something like:

```
#include <dbm.h>
..
/* assume dbm is open */
void load(pa)
    Path *pa;
{
    datum key, conts;

    key.dptr = pa->name;
    key.dsize =
        strlen(pa->name) + 1;
    conts.dptr = pa->path;
    conts.dsize =
        strlen(pa->path) + 1;

    store(key, conts);
}
```

I have neglected writing the `flags` field in the original structure out to disk. Notice how I have worried about ensuring that the null byte at the end of the string is copied to the data file. This just makes it easier when the data is returned into a program.

The `fetch` routine is given a key, looks for a dataset with that key and returns its value. So my lookup routine becomes:

```
Path *look(name)
    char *name;
{
    static Path ret;
    datum key, cont;

    key.dptr = name;
    key.dsize =
        strlen(name) + 1;

    cont = fetch(key);

    if (cont.dptr == NULL)
        return (Path *)0;

    ret.name = key.dptr;
    ret.path = cont.dptr;
    return &ret;
}
```

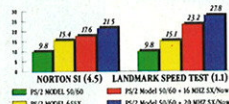
Here again the `datum` structure is passed by value rather than by reference. The

SX/Now! Processor Upgrade

Kingston
TECHNOLOGY EUROPE

Don't replace your system...
For Windows, for OS/2, for today's software,
you need 386SX processing power. With
Kingston's SX/Now!, you can replace the
processor on your 286 based computer - not
the entire system.

... upgrade your processor!
By combining a 16 or 20MHz 386SX
processor with 16KB of cache memory and
an onboard clock, SX/Now! increases system
performance by over 250% - without taking
an expansion slot!



Fully compatible
SX/Now! is fully compatible with your
system's hardware and software. And it is
covered by Kingston's five year warranty.

Cost-effective performance
So, to get the speed and
processing power your
software requires, choose
the cost-effective solution,
SX/Now! from Kingston.

Distributed in the UK by:-
DATRONTECH
COMMITTED TO MEMORY
the memory to remember

33 Grosvenor Road,
Aldershot,
Hants GU11 3DP
Tel: 0252 313155
Fax: 0252 341939

**UPGRADE
SOLUTION
CENTRE**

The Next Generation OF MATHS CO PROCESSORS

**40MHz
NOW
AVAILABLE**



**THE ULSI MATH-CO™
THE INTELLIGENT ALTERNATIVE**

- Lifetime warranty
- Advanced CMOS design for high speeds
25MHz SX available
40MHz DX available
- Unique Auto-sleep mode for optimum
battery life
- Fewer clock cycles for maximum
performance
- Hewlett Packard manufacturing technology
- 30 day money back guarantee

AT AN INTELLIGENT PRICE

**ULSI
SYSTEMS**

DATRONTECH
COMMITTED TO MEMORY

0252 313155

31-33 Grosvenor Road, Aldershot, Hants GU11 3DP.
Tel: 0252 313155 Fax: 0252 341939

		(£)
83C87SX	16MHz	121
	20MHz	130
	25MHz	CALL
	40MHz	CALL
83C87DX	16MHz	134
	20MHz	151
	25MHz	195
	33MHz	225
	40MHz	CALL

KINGSTON ORIGINAL MEMORY

WHY SPECIFY ANYTHING LESS THAN THE BEST?

Kingston
TECHNOLOGY EUROPE

**Manufacturer for compatible propriety
and commodity memory solutions for:**
IBM, COMPAQ, TOSHIBA, AST, APPLE, ZENITH, DEC, SUN,
PANASONIC, OKI, NEC, NCR, HP, EVEREX plus many others.

- Market leaders with over 300 products supplying 47 countries.
- Original manufacturer. Guaranteed consistency of quality, reliability and supply.
- All products carry lifetime warranty and are made in the USA.
- Extensive list of UK corporate and OEM endorsements, NSTL approvals.

- Largest R&D commitment to new product introduction.
- European manufacturer Q1 1992.
- **AUTHORISED DEALERS** enjoy added benefits of stock protection and co op advertising.

FIND OUT MORE TODAY - Call 0252 313155



**Seal
of Excellence**

DATRONTECH
COMMITTED TO MEMORY

Datrontech Ltd, 31-33 Grosvenor Road, Aldershot, Hants GU11 3DP.
Fax: 0252 341939.

Northern Office: Tel 061-928 8688 Fax 061-928 1724

MEMORY FOR LAPTOPS

Kingston
TECHNOLOGY EUROPE

"It's a poor sort of memory that
only works backwards," the Queen
remarked."

*Lewis Carroll
Through the Looking Glass*

We agree. Which is why Kingston manu-
factures a most versatile line of laptop
memory upgrades. Fully compatible mem-
ory for all the latest laptops from Apple,
Compaq, NEC, Texas Instruments, Toshiba,
Sharp, and Zenith. And our new laptop
memory offers the sort of margins that
keep your profits going forward, not back-
ward.

Manufacturers of
memory products for:

Apple • AST • AT&T

Canon • Compaq • DEC

Data General • Epson • HP

IBM • NCR • NEC • Silicon

Graphics • Sun • Toshiba • Zenith

United Kingdom
France
Germany
Italy
Spain

Scandinavia
Benelux
Switzerland
Austria
Portugal

These modules for
NEC ProSpeed 386,
Apple Macintosh Portable,
Compaq LTE 286, Zenith
SupersPORT SX and Toshiba
T3100SX are only a part of
Kingston's complete
laptop memory upgrade
line.



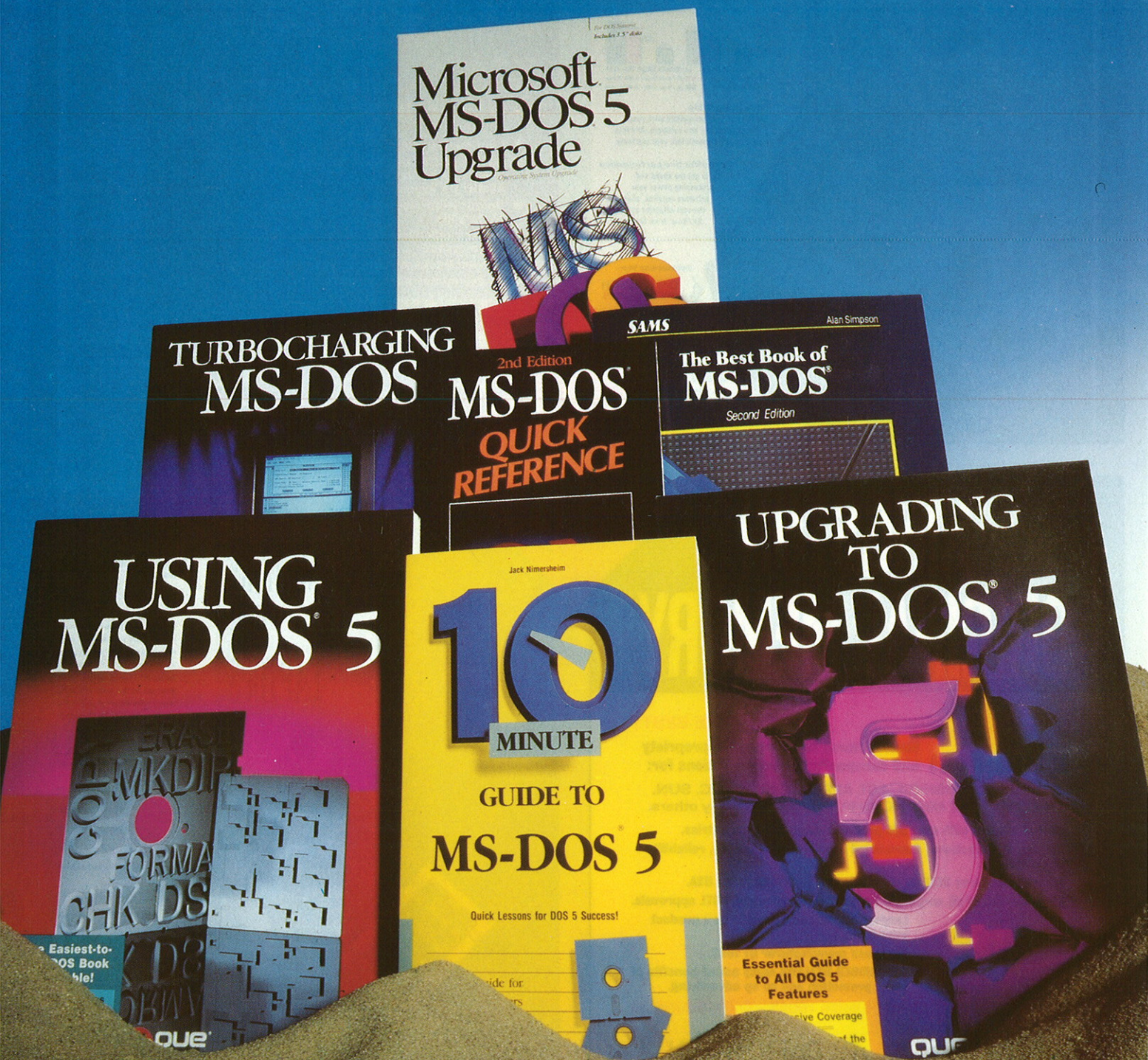
Distributed in the UK by:
DATRONTECH
COMMITTED TO MEMORY
the memory to remember

33 Grosvenor Road, Aldershot
Hants GU11 3DP
Tel: 0252 313155
Fax: 0252 341939

A member of the European PC Upgrade Forum

CIRCLE NO. 028

Raising MS-DOS



Even Higher

CIRCLE NO. 029

que

Essential MS-DOS 5 books from two of the leading publishers. QUE & SAMS lead the field with extensive ranges of computer books covering all subject areas. Available now at all good bookshops & computer stores. Call 021 706 6000 now for your nearest stockist. UK Distributor: Computer Bookshops Ltd, 30 Lincoln Road, Olton, Birmingham B27 6PA.

SAMS

fetch routine always returns something and will return a datum with zero data pointer if the item of the required key is not found.

Since the database is stored on a file, there needs to be some ability to alter the data. This is reasonably simple. The delete routine takes a key and removes the appropriate key/content pair from the file. The store routine will not complain if its action will result in the replacement of a key/content pair, it assumes that the code wants this to happen.

The final pair of routines allows you to step through all the data on the file. A typical loop is

```
for (ky = firstkey();
    ky.dptr != NULL;
    ky = nextkey(ky))
{
    /* do some work */
    con = fetch(ky);
    printf("%s\t%s\n",
        ky.dptr, con.dptr);
}
```

The firstkey routine returns the first key in the database and nextkey returns any later ones. The order that the keys are delivered are random.

As you can see, these routines are easy to use. They can only deal with one file at a time and this is a problem. Berkeley has supplied an alternative set of routines with the same functionality but a different interface. This set, called ndbm, allows more than one DBM database to be used simultaneously. These routines should be available on Berkeley derived systems and if they are available, you should use them. I have described the original set for simplicity.

Finally

One worry about using a routine suite like hsearch or ndbm is that your program stops being portable. You may come up against a system that doesn't support the routines that you have used as a crucial part of your program. I think that the standardisation efforts coupled with the move to System V Release 4 may help to alleviate these problems. In addition, public domain versions of the DBM code do exist.

Having routines like this in the library saves days of work and further days of worry about the failure of the new set of routines that someone has pulled out of thin air.

They allow you to make use of work that other people have done to generate algorithms based on good mathematics and computing practice.

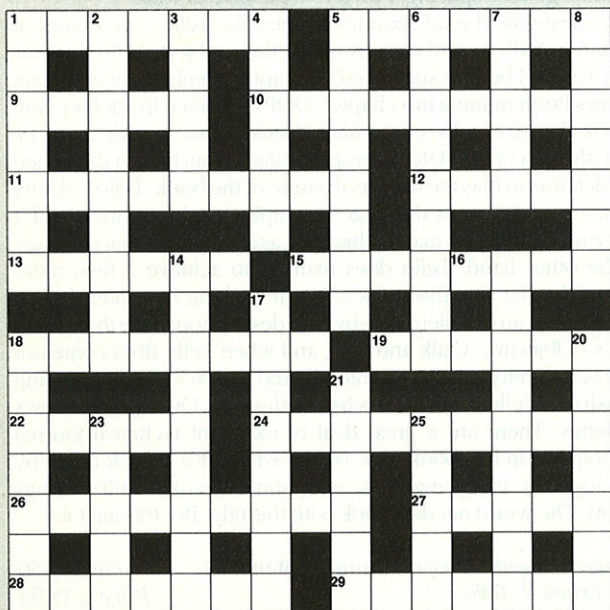
Reading

For details on sorting and hash tables look in the third volume of Knuth's 'The Art of Computer Programming', the volume is called 'Sorting and Searching'. The books are published by Addison-Wesley. The best recent reference for the mechanics of the DBM code is to be found in the Proceedings of the Winter Usenix, 1991 in Dallas, Texas. The paper is called 'A new hashing Package for UNIX' by Margo Seltzer and Oz Yigit. It describes some code that they have put in the public domain to provide both hsearch and dbm functionality.

EXE

Peter Collinson is a freelance consultant specialising in UNIX. He can be reached electronically as pc@hillside.co.uk (although your mailer might be happier to put the address the other way round) or by phone on 0227 761824.

JULY .EXEWORD



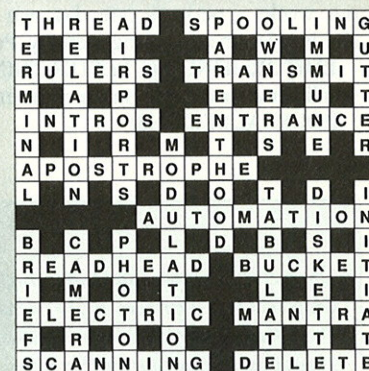
ACROSS

- 1 What a store! Really? (7)
- 5 Noise on the line may be evil (7)
- 9 In(n) to a procedure (5)
- 10 Program instruction for the police computer? (9)
- 11 Rat sound I make for long period (9)
- 12 Little sister uses aluminium to make hemp (5)
- 13 - and tally-ho! (6)

- 15 For instance THEN, ELSE, ENDIF (2, 6)
- 18 Checking up on making a search (8)
- 19 Flat crustacean with the same leg? (6)
- 22 Up to about 20kHz (5)
- 24 Step on (9)
- 26 Teachers with it in teacher's post (9)
- 27 We all know Eratosthenes' (5)
- 28 Favoured plants in Oz (7)
- 29 Tuneful items as sat on somehow (7)

DOWN

- 1 In the right form when entered may be... (7)
- 2 ... able to call again and again ... (9)
- 3 ... until comes round without a flame (5)
- 4 Learning time with fewer clothes (6)
- 5 Sort into sequence and make secret (8)
- 6 Old volume controls (9)
- 7 The important people at the end (5)
- 8 Drink, friend, completely! (7)
- 14 Input units for main groups of directors? (9)
- 16 Carry out with a tool (9)
- 17 Central flashes of understanding (8)
- 18 Small 24 (but a leap for Clive) (7)
- 20 Messes around to make it grey (7)
- 21 So spec becomes ranges of validity (6)
- 23 Data item once (5)
- 25 Micro-maker found in steps only once (5)



JUNE .EXEWORD

'EXEWORD' compiled by Eric Deeson

Books

Agents and actors in two differing guises this month.

Hex thespis

An office in East London. Morning. Enter scruffy JOURNALIST right, bearing book in swaddling sticky-tape. Yawns indolently. Sits down and picks at wrapping with broken biro. Face changes to shock.

JOURNO: 'Computers as Theatre?' What th-? [READS FURTHER] Ah. A book by Brenda Laurel. She's the one that calls herself an Interactivist, and edited a book last year called *The Art of Human-Computer Interaction*. [GRINS SMUGLY] Useful in parts, as I recall, but chock-a-block with California psychobabble. Computers as Theatre? [RUBS HANDS] Well, this review should be a cinch.

FX: 'PANNING.DOC' form being loaded from hard disk.

Enter EDITOR in black (cameo by Anthony Hopkins).

EDITOR: Not being...overly simplistic again, are we, Books Editor?

JOURNO: [SHADOW PASSING OVER FACE] Of course not. It will be arrant nonsense, I assure you. Listen to this [PICKS PASSAGE AT RANDOM] 'The design of an effective interface must begin with an analysis of what a person is trying to *do*, rather than with a metaphor or notion of what a screen should display'. Hmm. A sensible point. Must have stumbled on a good bit. [READS IN MORE DETAIL] 'The technical magic that supports representation in the theatre is behind the scenes... the representation is all there is. Existential WYSIWIG' [MUTTERS]

EDITOR: Better than you thought?

JOURNO: [GRUDGINGLY] Well, it chokes in my throat to say it, but ... Laurel does have a new angle on programming here. Her argument seems to be [EXPERTLY SKIM READING] that because programs are self-contained activities that humans have to enter into to understand, we should study the techniques that playwrights use to lull audiences into believing (and enjoying) the passage of plays. Users get annoyed with misfeatures in programs in much the same way as they do if Ophelia has a swift ciggie after Act III. It's not some absolute obedience to some desktop metaphor that's needed, but a sense of unity to the program. Programs don't work wrong - they work out of character.

EDITOR: I'm not convinced.

JOURNO: I'm not either, entirely - but it's an interesting theory, and Laurel does produce some earthy conclusions out of it for workaday coding. And, you know, there was something about her plea for greater romanticism in programming that touched me. Touched me deeply. [MISTS OVER] Perhaps we *can* strive for Aristotelian harmony within our works - perhaps one day, we programmers *will* make small children laugh and young men cry. Perhaps...

EDITOR: [FINDING BETTER THINGS TO DO] Yes, and perhaps you'll get your copy in on time this month.

[JOURNALIST SIGHS]



[CURTAIN]

EDITOR: [OFF] And no stupid gimmicks!

Title: *Computers as Theatre*

Author: Brenda Laurel

Price: £26.95

Pages: 209

Publisher: Addison-Wesley

ISBN: 0-201-51048-0

Tell all Tello

Rob Reiner's classic rockumentary 'This is Spinal Tap' has a number of sage thoughts on computer programming, but perhaps the most poignant in the context of Ernest R Tello's *Object-Oriented Programming For Windows* is the band's observation that 'It is a thin line between clever and stupid'. The parallel is this: OOP programming is, in the grand scheme of things, a young pursuit. Windows programming is even youthier. In both, the line between beginner and self-styled expert is thin. Moreover, nobody (yet) has all the bases covered. A few folk know Windows, some more few know about individual OOP languages, but no few nowhere know all the above. Programmers want a Master Guru - goodness knows we want one. But unless Messrs Petzold and Stroustrup are willing to indulge in some unorthodox genetic splicing, I do feel we'll have to wait.

In the meantime, broad-spanning books like Tello's are bound to have Thames Valley-sized chasms within their subject. Tello does (as the Von Trapps I believe suggested) attempt to climb every mountain - Windows Programming in a chapter, OOP explained in another - but in a book this size, issues are either dismissed so quickly as to be ignored (the tour of the SDK suffers particularly from this) or explained in such detail as to bias the universal nature of the book. Tello's clearly an Actor expert (most of the larger examples are in Actoric), and it would be more honest to market this book as a tutorial for that package.

On the other hand, Tello does manage to achieve a few, more modest goals, that give this book a Worth Looking At sticker. Principally, it includes an excellent side-by-side description of the three main OOPs Cs - Objective, Ctalk and C++, and when Tello does come out of his closet, a pretty good Actor tutorial. And Chapter 10 is a frustrating mishmash of excellent, but somewhat misdirected, OOP and Windows advice-lettes. There are a great deal of excellent technical journal articles trapped in this book. But, on the whole, it's a book made by putting together many manuals, with some (frankly dull) listings thrown in. The world needs a book with this title. But this ain't it.

Title: *Object-Oriented Programming for Windows*

Author: Ernest R. Tello

ISBN: 0-471-52957-5

Pages: 350

Price: £35.95

Publisher: Wiley

Book reviews by Mr Danforth O'Brien, BA Oxon.

Books Received This Month

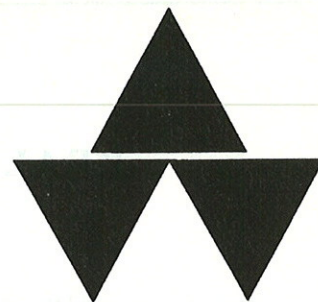
<i>Analysis of Algorithms and Data Structures</i> by Lech Banachowski	Addison-Wesley	ISBN: 0-201-41693-X	£22.95
<i>Graphical User Interfaces in C++</i> by Mark Goodwin	Pitman	ISBN: 1-55828-032-4	£22.50
<i>The C Book (2nd ANSI Edition)</i> by Mike Banahan et al	Addison-Wesley	ISBN: 0-201-54433-4	£21.95
<i>Developing Software for the User Interface</i> by Len Bass	Addison-Wesley	ISBN: 0-201-51046-4	£31.45
<i>Advanced X Window Applications Programming</i> by Johnson & Reichard	Pitman	ISBN: 1-55828-029-4	£26.95

NETWARE 386 PROGRAMMER'S GUIDE

Ralph Davis

Novell's NetWare, with over six million users, has become the networking standard for IBM PCs and compatibles. This book provides the NetWare 386 programmer with a complete, accessible, and practical guide to developing applications for the 80386 version of NetWare. It focuses on the specifics of NetWare Loadable Modules and Version 1.2 of the NetWare C Interface Library.

Summer/600pp/
0 201 57709 7/paper TBA



NETWARE SYSTEM INTERFACE TECHNICAL OVERVIEW

Novell

This is the official reference to Novell's C Network Compiler. It provides all the information you need to work with the NetWare operating system, allowing you to build carefully tailored applications with direct access to NetWare functions. All C programmers developing applications for a NetWare environment will find the book an indispensable resource.

Spring 1991/368pp/0 201 57027 0/paper £28.75

NETWARE TROUBLESHOOTING

Tricks and Techniques

Michael L Hader

This handy problem-solving guide offers invaluable tips and techniques to help NetWare system administrators get the most out of their systems. It covers all versions of NetWare, including NetWare 386. Topics discussed include managing the LAN, third party software and workstation optimization. With over 50 ready-to-run utility programs, this book is a must for all NetWare system administrators.

Book and 5" disk package/450pp/
0 201 57737 2/TBA

TCP/IP and NFS

Internetworking in a UNIX Environment

Michael Santifaller

This highly practical book is designed for non-network specialists to enable them to understand the techniques involved in networking with TCP/IP and NFS in a UNIX environment. It features a concise introduction to basic networking concepts and terminology, thorough coverage of TCP/IP and a clear account of NFS, its protocols and implementation.

Spring 1991/304pp/0 201 54432 6/hard £23.95

COMPUTER NETWORK STANDARDS: MAP and TOP

A Valenzano, C Demartini and L Ciminiera

This book aims to provide a basic understanding of computer network standards - MAP and TOP - in the field of factory and office automation. The main emphasis is on the illustration of the technical solutions adopted by MAP and TOP through a detailed presentation of the architecture of each individual protocol. The only book to present version 3.0, it has strong coverage of the latest developments in manufacturing message specifications, directory services and network management.

Summer 1991/464pp/0 201 41665 4/paper £24.95(tent)

ADDISON-WESLEY PUBLISHERS

Finchampstead Road, Wokingham, Berkshire RG11 2NZ

Tel: 0734 794000 Fax: 0734 794035

SOFTWARE

REAL-TIME, 'C'/UNIX, X-WINDOWS

Hayes & Stockport · to £22,000

Part of a successful British multinational, this company is committing £10M to the development of innovative intelligent control systems aimed at world-wide markets. Key new opportunities exist in the Software Development group for high-calibre graduate engineers with at least two years' relevant experience.

Initially, you would be involved with either Embedded Software or Graphical User Interface development, but opportunities exist to broaden your skills. Your involvement will be throughout the complete software development cycle, from concept to market.

For **Embedded Software**, a background in electronics is preferred, with real-time systems development experience. You must have ability with 'C' and assembler for Motorola and Intel microprocessors. Exposure to structured design techniques, SCADA,

industrial networks (eg ARCNET) or embedded kernels (eg OS-9) would be advantageous.

For **Graphical User Interface** development, fluency with 'C' and experience of using X-windows in a Unix-based RISC workstation environment is ideal, whilst familiarity with CASE tools, Open-Look or Motif would be an advantage.

The initiative is high-profile, non-military, technically exciting and very challenging. The purpose-built R & D environment is first-class, career development prospects are excellent and the remuneration package is attractive.


Please send your CV and salary details, quoting ref: 023/2/EXE, to: Michael Short, Personnel Consulting Services, Wyvern House, Station Road, Billingshurst, West Sussex RH14 9SE. Telephone: 0403 785120. Fax: 0403 784988.

Personnel Consulting Services

SEARCH SELECTION AND HUMAN RESOURCE CONSULTANTS

C:\Real-time\Comms\Broadcast

File Edit Search Work Options Help F1



Open a new Window on real-time control

An exciting opportunity to develop a new real-time control application using MS Windows. As part of a progressive software team, you will be involved in the specification and implementation of this project in a MS Windows environment, from its conception.

Position: Software Engineer with solid 'C' experience and knowledge of development under MS Windows.

Environment: You will be working within a small friendly team of software specialists, in the Development department at Philip Drake Electronics Ltd., located close to the A1 (M).

Applicants should send their CV to:
Shirley Nicholls, Personnel Manager,
Philip Drake Electronics Ltd.
The Hydeway,
Welwyn Garden City,
Herts, AL7 3UQ
Or Phone: (0707) 333866 (Fax: 371266)
for further details

QUESTOR

Computer Personnel

APPLICATIONS SOFTWARE DEVELOPMENT

Milton Keynes

£14 - £20k

With at least 2yrs' development experience, you will be an enthusiastic programmer who is interested in developing quality applications software utilising sophisticated development tools. Good 'C' skills essential, MS-DOS & Windows 3, OS/2 & Presentation Manager are desirable.

Contact PH

CAD-APPLICATIONS SOFTWARE ENGINEERS

S. Herts

£16k - £23k + Car

Highly regarded company established some 16 years, requires several additional applications development engineers to work on a 3D graphics product with colour imaging & near real-time walk throughs. Good C skills are essential, UNIX, Graphics or Fortran experience is beneficial.

Contact AH

CLIPPER PROGRAMMER

E. Herts

to £17k

This specialist software house designs and develops software for a range of major blue-chip companies, primarily in the financial sector. We are retained to recruit several programmers with experience of writing SUBSTANTIAL applications programs in Clipper and dBASE.

Contact PH

ORACLE DEVELOPMENT

N.Bucks

c. £17k

Specialist Hardware manufacturer with world-wide distribution is seeking a programmer with around 2 years' experience of low level Oracle and embedded SQL on any platform, for state of the art R&D environment.

Contact PH

PROGRAMMER

N. Bucks

£18-20k

Control Systems development with leading Systems House, using C on OS/2 platforms. Experience Presentation Manager or other window environments will be of interest and previous Control and Automation development work on PCs would be beneficial.

Contact PH

PROGRAMMER

S.Herts

£18-20k

Well known Systems House requires graduate Electronic Engineers with hardware design knowledge to work on Software development for Control and Automation Systems. Candidates should have sound C and assembler programming experience.

Contact PH

CUSTOMER SUPPORT ENGINEER

Thames Valley

£25k + Car

LAN manufacturer requires a Support Engineer with excellent presentation skills and knowledge of LAN equipment. Present products to clients; give technical support; Network Design; Customer Training; Advise Clients etc.

Contact AH

Plus many other positions throughout London, South East and the Home Counties. For further details of these or any other positions, send your CV in confidence or call Alan Hewson (AH) or Polly Hunter (PH) on:

Tel: 0462 438373

Fax: 0462 421272

Regal House

55 Bancroft

Hitchin

SG5 1LL

Evenings:

Alan Hewson on 0462 459338



Software Tools Developers

Southern Herts

to £25K

With unparalleled expansion plans, potentially the largest new commercial project in the U.K. and multi-national conglomerate financial backing, this company will rely heavily on the tools provided by its software development team. As part of this highly motivated team, you will be responsible for designing and developing advanced software tools using the latest relational and object orientated techniques. As a degree qualified software engineer with 2+ years C, C++, UNIX and ORACLE (or equivalent) experience this organisation offers career advancement second to none.

Future Systems

Berks

to £25K

In order to progress your career from Software Engineer to Team Leader and beyond you will need to demonstrate a sound technical knowledge of C, UNIX, OOD, RDBMS, and X-WINDOWS/MOTIF. You will be given the opportunity to improve your own skills, whilst being responsible for the development of advanced Network Management Systems. You will be assured exposure to emerging technologies, in order to determine their applicability to the systems you are developing and will be a vital technical member of a team dedicated to Network Management within this repeated commercial communications company.

GUI Designers

Europe/London to £32K + benefits

A unique opportunity to export your specialist GUI Design and Development skills to Europe. You will be presented with a working environment dedicated to exploiting leading edge technologies and promoting personal skills development. You will be responsible for designing complex applications and software systems, which will require an in-depth practical knowledge of one or more GUI standard: If you are degree qualified and can offer 3+ years experience of C, C++, UNIX, OS/2, PRESENTATION MANAGER, X-WINDOWS, MOTIF development in a formal environment then you could be playing a major part in the 1992 challenge of OPEN Europe.

C++, Windows

London

all levels to £30K + Car

A superb opportunity to enhance and develop your creative Software skills on innovative products and projects. A chance to make real use of your existing

skills as well as acquiring new ones on technically demanding commercial Software Systems within a leading results orientated IT company. With a good degree, experience of C, C++, Windows, OS/2, ORACLE, TOKEN RING LAN, UNIX and Structured Software development, the sky's the limit.

For more details

Contact Mike Jenkins for details on the above and many other vacancies on **0442 231691** days or 0442 253300 eves/wkends. Alternatively write to him at:

**Executive Recruitment Services,
Hempstead House, Selden Hill,
Hemel Hempstead. HP2 4LT**
or fax CV to 0442 230063.



FILE MANAGER

I · C · O · N · I · C · S

COMPUTER CONSULTANCY

Small friendly company specialising in electronic banking requires professional software engineers who wish to add consultancy to their range of professional skills.

We are looking for people with a minimum of two years' PC development experience. Knowledge of C, Windows and LANs is desirable.

If you enjoy a challenge and can respond positively to a stimulating and varied environment please apply in writing to:



Richard Measures
Iconics Ltd
40 Rosebery Ave
London EC1R 4RN

'C' PROGRAMMERS

Leading Products, Leading Company, Leading to.....

c£20,000 + Bonus + Benefits. Opportunities North and South.

As the Document Image Processing market advances, Image Systems Europe are widely acknowledged to be the No. 1 UK company, and are well positioned for further expansion in meeting the technical and commercial challenges of the future.

We now need additional programmers who have gained at least 12 months experience in 'C' and MS Windows within a PC environment. Familiarity with ORACLE/INGRES or other SQL based rdbms' is advantageous but not essential as comprehensive technical and product training is provided.

You'll be working in a small, friendly development team who recognise and employ new technologies to develop DIP systems across a diverse range of applications.

Excellent working conditions and starting salaries of up to £20,000 are on offer. Company benefits are also exceptional including: an INCENTIVE BONUS • FREE BUPA • NON CONTRIBUTORY PENSION and RELOCATION PACKAGE to a particularly attractive area of low cost quality housing.

'Enhance your image' by forwarding your CV (Ref. No. C804) to **Andre Evan Smith** at: **Aardvark Swift Associates Ltd, Westcourt House, 75-77 Station Street, SWINTON, South Yorkshire, S64 8PZ**

Alternatively, please contact **Debbie Leighton** on (0742) 420419 for further information.



IMAGE SYSTEMS EUROPE LTD

Low Level Software Engineer Surrey to £19k

Good low level skills, Assembler a must and MASM, TASM and BIOS ability are also essential requirements for one of the major software/hardware companies. PC and laptop development - you will be responsible, as part of a team, for developing a 'World Standard' product.

Systems Programmer Berks £20 - £30k + Bens

Micro to Mainframe ability; CICS, OS/2, MicroFocus' CICS option, PM, Windows, PCDOS and OS/2 EE skills are all a prerequisite for a really calibre job. C and Mainframe OS skills might also be advantageous. This is a good company, offering the potential for foreign work and travel, as well as excellent benefits and working conditions.

Programmer/Developer

Cambs/W London (2 positions) £16 - £20k neg

We are looking desperately for C/UNIX and X-Windows programmers who want their first or second job move into a rapidly expanding Software House. At least 1-2 years experience is essential, with additional expertise in PCs or workstations development an advantage. Lots of development, new products and a lucrative career path are all offered in return.

Software Engineer NE Surrey to £19k

Windows, C and some Excel experience up to macro level is desired. Different O/S and multi-platform work - massive developments planned, within a small team structure. Lots of different things happening in a growing Software House.

Technical Sales Support West London £20k

CAD technical and sales support, you will be preferably from an engineering background, as a user at some time or another, primarily with AutoCAD. You will be responsible for liaising between the technical and sales teams, and in helping to tailor solutions to the clients needs, so perhaps some coding history would be useful as well...

Software Installation Berks £20 - £25k

Ideally a graduate with a Technical Support background, primarily in MVS, but different platforms are an advantage. This position is a combination of software installation, technical support and capacity planning, with a very large Multi-National Corporation.

SQL Developer W London/Home Counties £16 - £25k

PC and/or UNIX platform, and a ability with C and Windows? This is a new vacancy in a small, quality software House utilising SQL as the main development language. At least 2 years experience is preferable.

Cobol Development Northern Counties neg up to £20k

Experienced commercial Cobol programming covering a variety of applications is required by this International Software House. The platform is not as important as the person, who must fit into the existing friendly team, but any AS/400 work would be an advantage. Cross training to this platform would in any case be provided, as would a relocation package if required.

Oracle/Ingres London to £25k

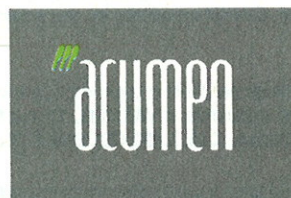
A major consultancy for specialist systems design is looking for RDB developers and designers. The skills base required is Oracle/Ingres/Sybase, with knowledge of SSADM. The preferred platform for this experience is not PC based but VAX, with the ideal candidate having worked on large projects in the past, within an established software house.

We also have a considerable number of Comms positions, both hardware and software orientated. These cover design, support and development. For further details, call or send a CV to Mike or Heather.



RONICOM
RECRUITMENT

5-7 Sedley Place (off Oxford Street), London W1R 1HH
Tel: 071 491 3640 Fax: 071 499 2546



Windows 3 Programmer

required for the development of

Voice & Image Processing Systems

Experience in C and Database
programming preferred

Salary £12K-£18K

dependant on experience

This is an excellent opportunity for
an ambitious individual to make
their mark in a fast growing young
business located in attractive
facilities near **Oxford**.

Please send resume marked
for the attention of the
Director of Development.

No agencies please

ACUMEN TECHNOLOGIES LTD
41 North Road · Islington · London · N7 9DP · England
Tel: 071-607 6766 · Fax: 071-607 6764

SOFTWARE PROFESSIONALS

SURREY

Leading transport consultancy seeking science, computing or mathematics graduates with **VAX/Fortran** or **'C'/UNIX** experience for new projects in autoguidance and route mapping.

to £18k

HANTS

Intelligent network developer seeking bright engineers with an interest in datacomms and at least 2 years' experience of **VAX/VMS**, **'C'** and **68000** assembler. Any of the following useful: **Windows**, **X.25**, **X.400**, **OSI**, **SNA** **X.75** or **C++**.

£15k-£22k

YORKS

Widely acknowledged as leaders in the **Document Image Processing** market, our client is seeking experienced **'C'** and **MS** or **X-Windows** software engineers. Any **Oracle/Ingres/SQL**, **UNIX**, **TCP/IP** or **Streams** useful. Bonus, BUPA, Pension and relocation offered.

to £20k

MIDDX

U.S. geoscience developer is seeking a scientific software programmer with proficiency in **'C'/UNIX**, to specify, design and implement a seismic data interface package. Any knowledge of **Graphics**, **X-Windows** or **VAX/Fortran** is useful.

£18k-£25k

S.HANTS/I.O.W

Varied and interesting work in the areas of **Process Control**, **CIM** and **Information Systems** if you have good **UNIX** and **'C'** experience, Waterfront location.

£neg.

BELGIUM or ITALY

International software house seeking **'C'** programmers with experience of two or more of the following: **MS-DOS**, **MS-Windows**, **Pascal**, **Oracle**, **UNIX** or **X-Windows**.

to £25k

BUCKS

Revolutionary emulation software product developer is seeking a **Software Project Manager** to direct and manage a number of specific projects, if you have good PC experience encompassing some of the following: **UNIX**, **DOS**, **Macintosh** or **X-Windows**.

£25k-£35k

SURREY

Analyst/programmers and programmers needed by leading accounting organisation if you have three of the following skills: **UNIX**, **'C'**, **Informix**, **Prolog**, **Oracle** or **ProC**.

£15k-£22k

W.COUNTRY

Systems Consultancy in the field of public utilities, government, legal and the pharmaceutical industries is keen to recruit good calibre graduates with some of the following skills: **'C'**, **UNIX**, **Paradox**, **RDBMS**, **CASE**, **SSADM**, **Prompt**, **Prince**, **Yourdon**, **MS-Windows**, **MS-DOS**, **OS/2**, **Presentation Manager**, **X.25**, **SNA**, **VAX** or **Networking**.

£18k - £25k

BERKS

The worlds leader in communications development is seeking Software Engineers with **VAX/VMS**, **Pascal** or **'C'** experience. Any of the following would be useful: **UNIX**, **Fortran**, **CCITT** or **X-Windows**.

£neg

HERTS

Leading force in the development of Printing Control Systems seek a real-time software engineer with most of the following: **'C'**, **Assembler**, **M68000**, **Z80**, **MS-DOS**, **OS/2** or **Hardware Interfacing**.

to £20k

BEDS

Exciting seismic exploration project requires real-time software engineers with most of the following: **'C'**, **UNIX**, **X-Windows**, **Motif** or **TCP/IP**.

£17k-£22k

HERTS

3D graphic tool developer running on **Apollo**, **Silicon Graphics** and **IBM RISC/6000** workstations is seeking **'C'/UNIX** programmers. Any experience of **Fortran**, **C++** or **CAD** useful. Excellent promotion prospects.

to £23k + Car

SURREY

Premier publishing software developer seeking talented software engineers with experience of **'C'/UNIX** or work on graphics projects utilising **SUN** and **Macintosh** workstations.

to £20k

E.ANGLIA

Internationally renowned comms developer seeking graduates/post-graduates for exciting projects in **Network Management**, **Videotex**, **Office Automation**, **Multi-media** and **PABX**. You should have at least two of: **VAX/VMS**, **'C'**, **UNIX**, **C++**, **MS-DOS**, **IKBS**, **M68000**, **Pascal**, **Oracle**, **DPNSS** or **SUN**.

£ Excellent

WILTS

Software Engineer and **Development Manager** sought for small teams developing **'C'** on **UNIX** under **MS-Windows**. Any other experience of **C++**, **X-Windows** etc useful.

£15k-£24k

For further information about these and MANY other opportunities either telephone

081 876-0102 / 081 392-1514

or write to: **ACUMEN SEARCH & SELECTION INTERNATIONAL**
London House, 42, Upper Richmond Road West, London SW14 8DD
Fax: 081 392-1518 Out of hours telephone 081 878-8206



S & S INTERNATIONAL

S & S International has been an established software house for over six years. We are famous for our Anti-Virus software and Data Recovery service. Our Chief Executive, Dr Alan Solomon, is a world renowned authority on computer viruses and security. Due to rapid expansion we have two new positions to fill.

PASCAL - DOS PCs

We need an experienced **PASCAL** programmer to control and coordinate development of our off-the-shelf product line and to help out with major in-house tasks. You should have at least two years' experience on PCs under **DOS**, and be used to working on large and complex projects. Low-level awareness is essential, and assembler experience would be even better. At the same time, familiarity with end-user needs is a must and some OOP exposure would also be of use.

Contact: Mrs Lindsay Holmes
S&S International
Berkley Court, Mill Street
Berkhamsted, Herts., HP4 2HB

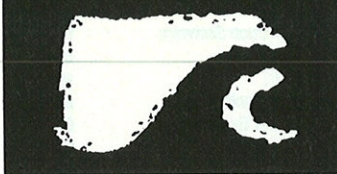
Data Recovery Specialist

We also need someone to join the team in our Data Recovery Centre. Our work focuses on PCs under **DOS** and **Novell**, and there is regular contact with other systems such as **UNIX** and **MacIntosh**. The position will bring experience in all aspects of recovery but will concentrate on the use and development of software based techniques. As well as being a competent programmer, you need to be an experienced **DOS** user with knowledge of several common PC applications. Above all, we need someone with a flair for creative problem solving in a high pressure environment.

Salary for both positions in negotiable up to £20K, depending on age and experience. If you are interested then please write with full CV, or telephone to arrange an interview. Direct applications will be given preferential consideration.

Tel: 0442 877877
Fax: 0442 877882

Software Care



Camden, London

(£15-20K)

C, DOS/UNIX Programmers

Database and GUI work
Integrated Hospital Systems

Software Care, a division of CHC (UK) Limited, develops information management and knowledge-based tools for the Nursing profession.

We are looking for programmers and analysts to join our small, highly motivated technical team. Having just completed a major development project, we are continuing development in multi-user database access, rule-based artificial intelligence, and communications.

You will have a flair for imaginative design of easy-to-use programs. While we will concentrate more on your ability to make our system work easily for busy nurses rather than your particular history, you will have a good grounding in C programming and some familiarity with CASE tools and structured design techniques.



Software Care Nursing Systems
Member of the CHC Group of Companies

For more information, send a brief CV and covering letter (in confidence) to Samantha Hilton-Johnson, Software Care Limited, Four Church Studios, Camden Park Road, London NW1 9AY.

VIGILANT CONSULTANCY SERVICES

PERMANENT

TECHNICAL SPECIALISTS

Currently require candidates with strong UNIX/XENIX OS/2 and Networking skills for Pre Sales and On-Site roles.

SALARY RANGE £15 - £20k + CAR
London & Surrey.

SOFTWARE SPECIALISTS

Current vacancies for candidates with skills in some if not all of these areas:

UNIX / C / ORACLE / INFORMIX / OS/2

Experience ranges from Programmers to Project Leaders.

SALARY RANGE £15 - £25k + BENS
London/Midlands/Home Counties

URGENT

Require Senior Software Engineer with experience of UNIX Device Drivers in SUN environment, for large Parallel Computing Project. Company working at the forefront of Technology, Transputers etc.

CONTACT

Adam Outen 8.30am to 7.00pm on 071-734-3499
or Fax your CV to 071-734-3306



ADVERTISERS INDEX

ADVERTISER	PRODUCT/SERVICE	CIRCLE	PAGE	ADVERTISER	PRODUCT/SERVICE	CIRCLE	PAGE
Addison Wesley	Book Publishers	030	89	Microcosm	Copy Protection Software	992	51
AET	Coherent UNIX	993	53	Microft	Security Software	027	83
AI International	Quintus Prolog	976	31	Nantucket	Clipper 5.01	962	5
Applied Logic Computing	Object Oriented Programming	024	81	Nu-Mega	Debugging Tools	964	8
Bits Per Second I	Graphics for Visual Basic	033	12	Pecan	Pecan Power System	037	57
Bits Per Second II	Graphics for dBASE	978	35	Pinna Electronics	Tools & Graphics	007	67
Bits Per Second III	Graphics Server Software	990	51	QA I	Windows Training	960	1FC
BL Computer Security	Software Protection	020	76	QAII	Software Testing	984	43
Blink Inc	Blinker fast linker	989	48	QAIII	C++ GUI Class Library	996	55
Borland	Turbo Pascal	972	26	QBS	Clipper Add-ons	975	30
Brent Communications	MAX copy PROTECTION	986	46	Real Time Software	CASE for Windows	977	32
Camel	Plotter Utilities	991	51	Recital	RDBMS/4GL for VAX & UNIX	969	20
CEBRA Communications	Multi VGA Adapters	979	35	Ridgeway	Computer Manuals/Binders	994	54
Clearsoft	Software Protection	031	OBC	Salford Software Marketing	Fortran for DOS & UNIX	999	61
Computer Bookshops	Book Publishers	029	86	SCL	Communications Boards	995	54
Comsec	Software Copy Protection	015	73	Sequiter	C Library	981	37
CTL	Copy Protection Hardware	005	66	Softlok International	Piracy Protection	022	78
Datrontech	Upgrade Packages	028	85	S/ware Construction Company	Development Tools	004	65
DES	Software Protection	023	80	S/ware Generation	Program Editor	971	25
Digital Equipment Company	Workstation applications	-	11	Solution Systems	Programming Editor	021	77
Digital Equipment Company II	Workstation applications	965	13	SQL Solutions	Systems Integration	987	46
D-Link	Network Operating System	985	44	System C	Application Generator	014	70
European Informix User Group	User Group	035	61	System Science I	Special Offer	008	67
Expert Systems	Prolog for Windows	016	73	System Science II/Lahey	FORTAN Compilers	012	69
Glockenspiel	C++ Software	968	19	System Science III	RM COBOL 85 Compiler	013	69
Grey Matter	Programming Tools	961	3	System Science IV	Development Tools	026	81
GW	C++ Embedded Software Design	980	35	System Star	dbms Four C	967	17
GW/Select	CASE Tools	032	12	TEXT Formatting Company	TEX/LATEX Software	025	81
Intasoft	Software Management System	973	27	The Data Business	Software Protection	019	76
IXI	UNIX Software and Training	009	68	The Minix Centre	UNIX	038	57
JPI	C++ Compilers	963	7	Total Control	Disk Duplication	997	57
Kibworth	C, C++, Windows Training	034	61	Univision	UNIX Development Systems	017	73
Korala	Directory Utility	006	66	USA Software	Programming Tools	970	23
LBMS	Multi-user CASE	983	41	User Friendly	Software Copy Protection	003	64
Linx	All UNIX Systems	018	75	Vogon	Data Interchange Software	101	69
LPA	AI/KBS/OOPS Software	011	69	Xoren	Communication Software	982	44
Magnifeye	Software Protection Device	974	29	Zortech I	Multi Plat C++ Comps	998	59
Matchpoint Design	Software Packaging	966	17	Zortech II	C++ Video Tutorial	002	63

STOB - A Chance to Meet You

Stob is seeking alternative employment.

I arrive at Dismalton, Essex with three quarters of an hour in hand (having set off well in advance to cater for Network South East timetable eccentricities), so I purchase an unwanted coffee and perch uncomfortably in the station caff, pretending to read newspaper, so as not to turn up at WayAhead Computers Ltd too early. Sitting here in my interview gear, trains rushing past while I'm indulging in treachery, feel like Celia Johnson in *Brief Encounter*. Mustn't think about *Brief Encounter*, not when I'm this nervous. Might break into *Brief Encounter* dialogue mid-interview, 'Oh darling, I can't tell you how heppy and carefree and gay I feel.' Worse yet, I might start humming Rachmaninoff's Second Bloody Piano Concerto. Too late. I **am** humming RSBPC.

Let's go now. I'll only be thirty-five minutes early. That's not early, that's just really prompt. Surely this hideous concrete and aluminium monstrosity is not the place? Come on Verity, wrong attitude, think positive.

Hello Miss Reception Girl, yes Mr Shuffle is expecting me; look, there I am in the diary 11:15 - Verity Stab. Take a seat along-

side three other candidates, engineer a sincere smile. 'Yes, the weather is grim isn't it?' Why is this man making conversation? I don't want to talk to him. I don't want to talk to anybody, least of all him. I want him to explode.

I bet I get called in last. I'll have to sit here, getting ever lonelier, the last green bottle on the wall, while all the others get in there and make their fiendish Good Impressions and witty light conversation, coming out smiling and shaking hands and pencilling in a provisional starting date. I won't get it. Everybody knows the last candidate never gets it.

'Miss Stab, will you come through now?' Oh God, I'm first. Why am I first? They're just getting the trash over with first, that's why. Oh God, the back of my neck has gone all cold. Now, up you get, don't trip over the high heels of your interview shoes, well done, another warm smile, stop humming Rachmaninoff, here we go.

'Nice offices you have here.' Oh, good shot Verity! Straight out of the Interviewee's Book of Ingratiating Phrases. We're on our way now. What's this? He's testing me on

assembler mnemonics. Well there's a low-down trick. I'm glad I'm not going to be working for you, Mr Shuffle, because I don't like people who play tricks like this. What do you think reference manuals are for, propping up the uneven legs of your horrible office furniture?

'HOCCE? Er, yes, that means HOLD Carry Count Flag, I think.' Or Halt On Computer Caught Fire. Beep-bi-beep-beep-beep, and at the end of that round, Miss Stob, you have scored two points. You passed on 15... I wish I wasn't here. I wish I was back in my dull little, safe little office, curled up with a cold coffee and Microsoft C.

'Music? Yes, well I really like Rachmaninoff.' Rachmaninoff? *Rachmaninoff*? Just as well he didn't ask you your favourite author, you'd probably have said Shakespeare. Aha, it's over now is it; no, thank *you* for *your* time, you greaseball, I'll see myself out, bye-bye Mr Shuffle, bye-bye Other Candidates, bye-bye Miss Reception Girl, there's the door, try not to run, aaaah - fresh air and liberty.

Well, how do you think I did?

EXE

*** EXCEPTIONAL PEOPLE - EXCEPTIONAL GROWTH ***

Formed in 1985, QA Training Ltd is now an acknowledged leader in the field of technical training with an international client list including many of the major corporates, software development houses and computer manufacturers. We are committed to quality and excellence in everything we do - and our reputation has been built firmly on the exceptional calibre of our staff.

We are seeking to expand our team of Training Consultants - positions which will offer you an unique blend of teaching, learning and consultancy - and currently have the following positions available:

*** GROUP LEADER - NETWORK SYSTEMS**

You would be responsible for i) leading a team of up to 8 Instructors both by example in the training room and by on the job coaching, counselling and advising ii) providing the technical focus and authority for the Network Systems training area. Can you demonstrate skills in the following areas:

- Real world experience and broad understanding across LANS/COMMS/CONNECTIVITY spectrums
- Specific expertise in one of the above
- IBM PCLP and IBM OS/2 LAN Server
- PC to PC Comms
- PC to Mainframe Connectivity
- SNA
- APPC
- TCP/IP
- Knowledge of operating systems such as DOS, OS/2 and UNIX
- Man management experience

*** WINDOWS DEVELOPMENT SPECIALISTS**

Courses and seminars with which you would be involved will depend on your experience, but would include ■ Microsoft Windows Programming ■ Advanced Windows ■ Troubleshooting Windows Designing Windows Applications. Can you demonstrate specific technical skills in

- Programming in WINDOWS - extensive experience
- C programming - at least 2 years' experience
- Exposure to design methodology
- Knowledge of Presentation Manager and/or OS/2
- Real world development experience
- Ideally project management experience

*** UNIX/AIX SENIOR CONSULTANT**

Where you would be teaching designated courses such as UNIX Advanced Shell Programming ■ UNIX System Programming ■ UNIX Communications ■ UNIX Device Drivers. Can you demonstrate the following skills:

- At least 5 years' experience in a technical computing environment, of which the last 3 will have been dedicated to UNIX
- Working knowledge of C programming under UNIX
- Excellent knowledge of UNIX at the shell interface level
- Working knowledge of several of the following: Shell programming; system calls; installation & administration.
- Working knowledge of one or more of the following would be desirable: TCP/IP; NFS; BNU; X Windows; OSF Motif; UNIX Device Drivers.

For any of the above vacancies, please send your Curriculum Vitae, quoting your salary requirements to:

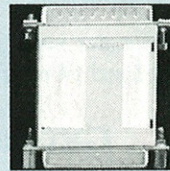
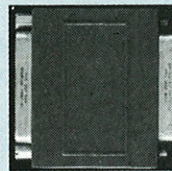
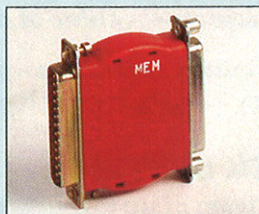
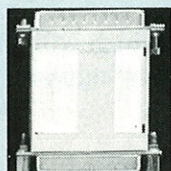
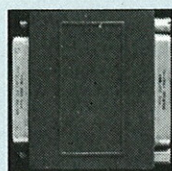
Melinda Bishop
Personnel Manager
QA Training Ltd.,
Cecily Hill Castle
Cirencester,
Glos. GL7 2EF

- First Class Salary
- Car plus Benefits
- Superb Location



SOFTWARE PROTECTION!

Can You Spot The Difference?



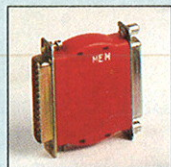
- ☒ High-level security keys
- ☒ Assembler-based, customer specific, encrypted interrogation routines
- ☒ Over 70 languages supported
- ☒ Compatibility - due to 8 years experience & 600,000+ keys sold
- ☒ Reliable on-going support
- ☒ Physically unique keys for each customer
- ☒ MS-DOS, OS/2, UNIX, XENIX, WINDOWS 2 & 3, "MACINTOSH"
- ☒ Transparent operation with most peripherals on market
- ☒ Free language updates
- ☒ Parallel, Serial, Mac ports

OVER 3,100 CUSTOMERS WORLDWIDE HAVE!



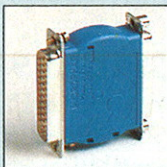
Electronic Key

The ideal device for identically produced software packages. Uniquely wired with customer code and a software code. Uses Assembler based program, decryption interface and random values.



1 Word Memory Key

Custom hardware wiring allows the developer total control over information stored in the key. 2 bytes of memory allows several packages to be protected with just the one key.



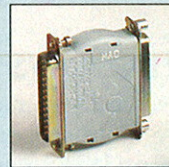
31 Word Memory Key

For multiple software protection schemes. 31 words of 16 bits of non-volatile dynamically programmable memory. Its capacity to store information provides virtually limitless power. Flexible protection scheme can be modified on-site during operation of software package.



Micro Processor Key

Provides the ultimate in software security. Not tied to any language or O/S. 8 bit microprocessor powers from RS-232 level. Requires no power supply. For PC terminals, minis, & others using RS 232 C comms. Used on workstations. This key is effectively a computer.



Macintosh Key

Extremely powerful & customised protection for the Mac. 31 words available for random storage. Providing unequalled protection the Macintosh Memory Key connects to the SCSI 25 pin port and operates transparently.

MICROPHAR... The European Standard!

For a no-nonsense informative discussion on how our dongles can better protect your profits, please call us on:

091-378 9191

CLEARSOFT

CLEARSOFT, Littleburn Ind. Estate,
Langley Moor, Durham DH7 8HG.
Telephone: (091) 378 9191 Fax: (091) 3789393

Please send me full details on your protection systems.

Name:

Position:

Address:

Telephone:

Fax:

Signature:

Clearsoft, Littleburn Ind Est,
Langley Moor, Durham, DH7 8HG
Tel: 091 3789191 Fax: 091 3789393

EXE 7/91